



TUGAS AKHIR - KI141502

REALITAS VIRTUAL SISTEM NAVIGASI PADA LINGKUNGAN VIRTUAL MALL DENGAN MENGGUNAKAN NAVMESH DAN GOOGLE CARDBOARD

WAHYU WIDYANANDA
NRP 5112100159

Dosen Pembimbing
Dr. Darlis Herumurti, S.Kom., M.Kom.
Anny Yuniarti, S.Kom., M.Comp.Sc.

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2016

(Halaman ini sengaja dikosongkan)



FINAL PROJECT- KI141502

VIRTUAL REALITY NAVIGATION SYSTEM IN VIRTUAL MALL ENVIRONMENT USING NAVMESH AND GOOGLE CARDBOARD

WAHYU WIDYANANDA
NRP 5112100159

Advisor
Dr. Darlis Herumurti, S.Kom., M.Kom.
Anny Yuniarti, S.Kom., M.Comp.Sc.

DEPARTMENT OF INFORMATICS
FACULTY OF INFORMATION TECHNOLOGY
SEPULUH NOPEMBER INSTITUTE OF TECHNOLOGY
SURABAYA
2016

(Halaman ini sengaja dikosongkan)

LEMBAR PENGESAHAN

REALITAS VIRTUAL SISTEM NAVIGASI PADA LINGKUNGAN VIRTUAL MALL DENGAN MENGUNAKAN NAVMESH DAN GOOGLE CARDBOARD

Tugas Akhir

Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer
pada

Rumpun Mata Kuliah Interaksi Grafika dan Seni
Program Studi S-1 Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember

Oleh:

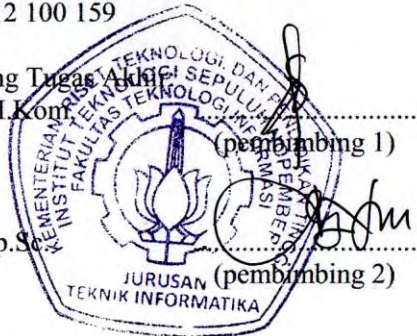
WAHYU WIDYANANDA

NRP. 5112 100 159

Disetujui oleh Dosen Pembimbing Tugas Akhir

Dr. Darlis Herumurti, S.Kom., M.Kom.

NIP: 19771217 200312 1 001



(pembimbing 1)

Anny Yuniarti, S.Kom., M.Comp.Sc.

NIP: 19810622 200501 2 002

(pembimbing 2)

SURABAYA

JULI, 2016

(Halaman ini sengaja dikosongkan)

REALITAS VIRTUAL SISTEM NAVIGASI PADA LINGKUNGAN VIRTUAL MALL DENGAN MENGUNAKAN NAVMESH DAN GOOGLE CARDBOARD

Nama Mahasiswa : Wahyu Widyananda
NRP : 5112 100 159
Jurusan : Teknik Informatika FTIf-ITS
Dosen Pembimbing I : Dr. Darlis Herumurti, S.Kom., M.Kom.
Dosen Pembimbing II : Anny Yuniarti, S.Kom., M.Comp.Sc.

ABSTRAK

Bagi setiap orang yang baru mengunjungi suatu mall sebagian besar akan merasa kesulitan untuk mencari lokasi dari tempat yang ingin dituju pada mall tersebut. Dengan bantuan beberapa navigasi yang ada pada mall seperti penunjuk arah, papan informasi, serta pusat informasi masih belum menyelesaikan permasalahan secara maksimal. Oleh karena itu, dibutuhkan sebuah aplikasi sebagai alternatif dari solusi yang ada dengan memanfaatkan teknologi yang ada.

Realitas virtual adalah salah satu solusi untuk mengatasi masalah tersebut. Berbagai tempat dalam lingkungan mall akan dimodelkan dalam bentuk 3D menyerupai kondisi sebenarnya kemudian membuat pengguna dapat berinteraksi di dalamnya. Diperlukan sebuah sistem navigasi dalam lingkungan virtual agar pengguna tidak kebingungan dalam melakukan pencarian lokasi dalam lingkungan virtual. Pada Tugas Akhir, penulis membangun aplikasi realitas virtual sistem navigasi pada lingkungan virtual mall dengan NavMesh dan Google Cardboard. Aplikasi ini akan melakukan navigasi dengan mencari rute terdekat untuk menuju ke destinasi.

Uji coba pada Tugas Akhir ini terdiri dari pengujian fungsionalitas aplikasi, serta pengujian aplikasi terhadap pengguna. Dari hasil pengujian aplikasi yang telah dilakukan oleh penguji, dapat disimpulkan bahwa aplikasi yang dibuat lebih

nyaman digunakan oleh pengguna yang memiliki pengalaman menggunakan Google Cardboard. Dari hasil kuisisioner yang telah diajukan kepada penguji, dapat disimpulkan bahwa aplikasi sudah nyaman, mudah digunakan, berguna, dan cukup immersive. Dengan dikembangkannya aplikasi ini diharapkan bisa membantu pengguna dalam melakukan pencarian lokasi pada mall, baik secara nyata maupun dalam lingkungan virtual.

Kata kunci: Realitas Virtual, Navigasi, NavMesh, Google Cardboard

VIRTUAL REALITY NAVIGATION SYSTEM IN VIRTUAL MALL ENVIRONMENT USING NAVMESH AND GOOGLE CARDBOARD

Student Name : Wahyu Widyananda
NRP : 5112 100 159
Major : Teknik Informatika FTIf-ITS
Advisor I : Dr. Darlis Herumurti, S.Kom., M.Kom.
Advisor II : Anny Yuniarti, S.Kom., M.Comp.Sc.

ABSTRACT

For every people who have barely visited the mall will find it difficult to locating a place in mall. With the help of the navigation in the mall such as directions, information boards, as well as an information center still does not fix the problem optimally. Therefore, it takes an application as an alternative to existing solutions by leveraging existing technology.

Virtual reality is one of the solutions to resolve the issue. Various places in the mall environment will be modeled in 3D resemble actual conditions then allowing users to interact in it. Needed a navigation system in a virtual environment so users are not confused in the search locations in the virtual environment. In the Final, the authors build a virtual reality applications navigation system in a virtual environment with NavMesh mall and Google Cardboard. This application will navigate by finding the shortest route to reach the destination.

Tests on this final project consists of testing the functionality of the application, as well as testing the application to the user. From the results of testing of applications that have been made by the examiner, it can be concluded that the application is made more convenient to use by users who have experience using Google Cardboard. From the results of questionnaires that have been submitted to the examiner, it can be concluded that the application is convenient, easy to use, useful, and quite immersive. With the development of this application is

expected to assist users in searching the location of the mall, both real and virtual environments.

Keywords: Virtual Reality, Navigation, NavMesh, Google Cardboard

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

Segala puji dan syukur, kehadiran Allah Subhanahu wa ta'ala yang telah memberikan rahmat dan hidayah-Nya sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul “Realitas Virtual Sistem Navigasi Pada Lingkungan Virtual Mall Dengan Menggunakan NavMesh dan Google Cardboard”.

Pengerjaan Tugas Akhir ini adalah momen bagi penulis untuk mengeluarkan seluruh kemampuan, hasrat, dan keinginan yang terpendam di dalam hati mulai dari masuk kuliah hingga lulus sekarang ini, lebih tepatnya di jurusan Teknik Informatika Institut Teknologi Sepuluh Nopember Surabaya.

Dalam pelaksanaan dan pembuatan Tugas Akhir ini tentunya sangat banyak bantuan yang penulis terima dari berbagai pihak. Melalui lembar ini, penulis ingin secara khusus menyampaikan ucapan terima kasih kepada:

1. Allah SWT yang telah melimpahkan rahmat, hidayah, dan inayah-Nya sehingga penulis mampu menyelesaikan Tugas Akhir dengan baik.
2. Keluarga penulis, Bapak Prayitno, Ibu Sri Rulianah, adik Fika Rizkyanti, dan juga keluarga yang tidak dapat penulis sebutkan satu per satu yang telah memberi dukungan moral dan material serta doa untuk penulis.
3. Bapak Dr. Darlis Herumurti, S.Kom., M.Kom. selaku dosen pembimbing I yang telah memberikan bimbingan dan arahan dalam pengerjaan tugas akhir ini.
4. Ibu Anny Yuniarti, S.Kom., M.Comp.Sc. selaku dosen pembimbing II yang telah memberikan bimbingan dan arahan dalam pengerjaan tugas akhir ini.
5. Teman-teman laboratorium Interaksi Grafis dan Seni yang selalu memberi dorongan dan inspirasi kepada penulis.

6. Teman-teman kontrakan BME E-45 (Satriya, Metana, Ifan, Mamin, Ardhya, Hafidh, Fandy, Alief, Bovy) yang telah menemani hidup penulis selama tiga tahun, berbagi ilmu, menjaga kebersamaan, dan memberi motivasi kepada penulis.
7. Teman-teman Hedon Hedin (Hanif, Fariz, Risa, Ariani, Ibor, Firqin) yang senantiasa memberikan dorongan, inspirasi, dan hiburan tersendiri bagi penulis.
8. Teman-teman Mahasiswa Teknik Informatika 2012 yang telah berjuang bersama-sama selama menempuh pendidikan di Jurusan ini.
9. Serta semua pihak yang belum sempat disebutkan satu per satu yang telah membantu terselesaikannya Tugas Akhir ini, penulis mengucapkan terima kasih.

Penulis telah berusaha sebaik mungkin dalam menyusun Tugas Akhir ini, namun penulis mohon maaf apabila terdapat kekurangan, kesalahan maupun kelalaian yang telah penulis lakukan. Kritik dan saran yang membangun dapat disampaikan sebagai bahan perbaikan selanjutnya.

Surabaya, Juli 2016

Wahyu Widyananda

DAFTAR ISI

LEMBAR PENGESAHAN.....	v
ABSTRAK	vii
ABSTRACT	ix
KATA PENGANTAR.....	xi
DAFTAR ISI	xiii
DAFTAR GAMBAR	xvii
DAFTAR TABEL	xxi
DAFTAR KODE SUMBER	xxiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah	3
1.3 Batasan Masalah	3
1.4 Tujuan.....	4
1.5 Manfaat.....	4
1.6 Metodologi	4
1.7 Sistematika Penulisan	6
BAB II TINJAUAN PUSTAKA.....	9
2.1 Realitas Virtual	9
2.2 Google Cardboard	10
2.3 Blender	11
2.4 Pemodelan Tiga Dimensi (3D)	11
2.5 Unity 3D	11
2.6 Microsoft Visual Studio	12
2.7 Navmesh	12
2.8 Line Renderer	13
BAB III ANALISIS DAN PERANCANGAN.....	15
3.1 Analisis	15
3.1.1 Analisis Permasalahan	15
3.1.2 Deskripsi Umum Aplikasi	17
3.1.3 Analisis Kebutuhan.....	17
3.1.4 Analisis Pengguna	20
3.2 Perancangan Aplikasi	20

3.2.1	Perancangan Skenario Kasus Penggunaan.....	20
3.2.2	Perancangan Arsitektur Aplikasi	28
3.2.3	Perancangan Lingkungan Virtual Mall	28
3.2.4	Perancangan Area Navigasi	30
3.2.5	Perancangan Objek Pengguna.....	30
3.2.6	Perancangan Informasi Ruangan	32
3.2.7	Perancangan Sistem Navigasi	32
3.2.8	Perancangan Skenario Tutorial	36
3.2.9	Perancangan Skenario Menu Utama	37
BAB IV IMPLEMENTASI		39
4.1	Lingkungan Implementasi	39
4.1.1	Perangkat Keras	39
4.1.2	Perangkat Lunak	39
4.2	Implementasi Lingkungan Virtual Mall	40
4.3	Implementasi Area Navigasi.....	44
4.4	Implementasi Objek Pengguna	47
4.4.1	Implementasi Integrasi Google Cardboard dengan Unity	47
4.4.2	Implementasi Kontrol Pengguna.....	50
4.4.3	Implementasi Objek Sebagai <i>NavMesh Agent</i>	51
4.5	Implementasi Informasi Ruangan	52
4.6	Implementasi Sistem Navigasi	54
4.6.1	Implementasi Menu Navigasi	54
4.6.2	Implementasi Fitur Navigation(Path).....	59
4.6.3	Implementasi Fungsi Eskalator.....	61
4.6.4	Implementasi Fungsi Lift.....	64
4.6.5	Implementasi Fitur <i>Navigation(Auto)</i>	67
4.7	Implementasi Skenario Tutorial	69
4.8	Implementasi Skenario Menu Utama	71
BAB V PENGUJIAN DAN EVALUASI		73
5.1	Lingkungan Pengujian	73
5.2	Pengujian Aplikasi.....	73
5.2.1	Pengujian Fungsionalitas	73
5.2.2	Pengujian Aplikasi Terhadap Pengguna	93
5.3	Evaluasi	96

5.3.1	Evaluasi Pengujian Fungsionalitas	96
5.3.2	Evaluasi Pengujian Aplikasi Terhadap Pengguna ...	97
BAB VI KESIMPULAN DAN SARAN.....		105
6.1.	Kesimpulan.....	105
6.2.	Saran.....	106
DAFTAR PUSTAKA.....		107
A.	LAMPIRAN A KUISIONER	109
BIODATA PENULIS.....		139

(Halaman ini sengaja dikosongkan)

DAFTAR GAMBAR

Gambar 2.1 Permainan dengan Memanfaatkan Teknologi Realitas Virtual.....	9
Gambar 2.2 Gambaran Implementasi Navmesh pada Unity	13
Gambar 2.3 Implementasi Line Renderer pada Unity.....	14
Gambar 3.1 Diagram Kasus Penggunaan	21
Gambar 3.2 Rancangan Sederhana Arsitektur Aplikasi	28
Gambar 3.3 Denah East Coast Center – Pakuwon Lantai 1	29
Gambar 3.4 Denah East Coast Center – Pakuwon Lantai 2	29
Gambar 3.5 Denah East Coast Center – Pakuwon Lantai 3	30
Gambar 3.6 Rancangan Antarmuka Pengguna.....	31
Gambar 3.7 Rancangan Antarmuka Menu Pilihan Navigasi.....	33
Gambar 3.8 Rancangan Antarmuka Pilihan Rute Navigasi	33
Gambar 3.9 Rancangan Antarmuka Pilihan Destinasi	34
Gambar 3.10 Rancangan Tampilan <i>Virtual Path</i>	35
Gambar 3.11 Rancangan Tampilan Informasi Ruangan	35
Gambar 3.12 Rancangan Antarmuka Skenario Menu Utama	37
Gambar 4.1 File Add-on Archimesh	40
Gambar 4.2 Tampilan ‘ <i>User Preferences</i> ’ sebelum terinstall add-on Archimesh	41
Gambar 4.3 Tampilan ‘ <i>User Preferences</i> ’ setelah ter-install add-on Archimesh	41
Gambar 4.4 Pembuatan ruangan pada Blender	42
Gambar 4.5 Tampilan ruangan setelah dimasukkan ke dalam Unity.....	43
Gambar 4.6 Tampilan Lingkungan Virtual Mall Setelah Diberi Pencahayaan.....	44
Gambar 4.7 Tampilan Tab ‘Object’ pada Window Navigation ..	45
Gambar 4.8 Tampilan Tab ‘ <i>Bake</i> ’ pada <i>Window Navigation</i>	46
Gambar 4.9 Tampilan <i>Navigation Mesh</i> yang telah terbentuk....	46
Gambar 4.10 Tampilan Objek Pengguna	47

Gambar 4.11 Proses Impor <i>Package</i> Integrasi Google Cardboard ke Unity	48
Gambar 4.12 Hasil Proses Impor <i>Package</i> Integrasi Google Cardboard	49
Gambar 4.13 Hasil Proses Penambahan Kamera Cardboard.....	49
Gambar 4.14 Hasil Proses Pendefinisian Objek Pengguna Sebagai <i>NavMesh Agent</i>	52
Gambar 4.15 Tampilan Informasi Ruangan	53
Gambar 4.16 Tampilan Panel Penanda Ruangan	54
Gambar 4.17 Tampilan Menu Navigasi	55
Gambar 4.18 Tampilan Menu Pemilihan Rute	57
Gambar 4.19 Tampilan Menu Pemilihan Destinasi.....	58
Gambar 4.20 Tampilan Garis Navigasi	60
Gambar 4.21 Tampilan Garis Navigasi Melewati Eskalator	64
Gambar 4.22 Tampilan Skenario Tutorial	69
Gambar 4.23 Tampilan Menu Utama	71
Gambar 5.1 Kondisi Awal Pengujian Mencari Menu Navigasi ..	74
Gambar 5.2 Kondisi Akhir Pengujian Mencari Menu Navigasi..	75
Gambar 5.3 Kondisi Awal Pengujian Menggunakan Fitur <i>Navigation (Path)</i> Dalam Lantai yang Sama.....	76
Gambar 5.4 Kondisi Akhir Pengujian Menggunakan Fitur <i>Navigation (Path)</i> Dalam Lantai yang Sama.....	76
Gambar 5.5 Kondisi Awal Pengujian Mengikuti <i>Path</i> yang Terbentuk Untuk Menuju Lokasi Dalam Lantai yang Sama	78
Gambar 5.6 Kondisi Akhir Pengujian Mengikuti <i>Path</i> yang Terbentuk Untuk Menuju Lokasi Dalam Lantai yang Sama	78
Gambar 5.7 Kondisi Awal Pengujian Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator.....	80
Gambar 5.8 Kondisi Akhir Pengujian Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator.....	80
Gambar 5.9 Kondisi Awal Pengujian Mengikuti <i>Path</i> Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Eskalator	82

Gambar 5.10 Kondisi Akhir Pengujian Mengikuti <i>Path</i> Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Eskalator.....	82
Gambar 5.11 Kondisi Awal Pengujian Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift.....	84
Gambar 5.12 Kondisi Akhir Pengujian Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift.....	84
Gambar 5.13 Kondisi Awal Mengikuti <i>Path</i> Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Lift	86
Gambar 5.14 Kondisi Akhir Mengikuti <i>Path</i> Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Lift	86
Gambar 5.15 Kondisi Awal Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Sama	88
Gambar 5.16 Kondisi Akhir Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Sama	88
Gambar 5.17 Kondisi Awal Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator.....	90
Gambar 5.18 Kondisi Akhir Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator.....	90
Gambar 5.19 Kondisi Awal Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift.....	92
Gambar 5.20 Kondisi Akhir Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift.....	92
Gambar A.1 Lembar Kuisisioner Dimas Riskahadi.....	109
Gambar A.2 Lembar Kritik dan Saran Dimas Riskahadi	110
Gambar A.3 Lembar <i>Usability Testing</i> Dimas Riskahadi	111
Gambar A.4 Lembar Kuisisioner Hafieludin Yusuf Rizana	112

Gambar A.5 Lembar Kritik dan Saran Hafieludin Yusuf Rizana	113
Gambar A.6 Lembar <i>Usability Testing</i> Hafieludin Yusuf Rizana	114
Gambar A.7 Lembar Kuisioner Fajar Setiawan	115
Gambar A.8 Lembar Kritik dan Saran Fajar Setiawan.....	116
Gambar A.9 Lembar <i>Usability Testing</i> Fajar Setiawan.....	117
Gambar A.10 Lembar Kuisioner Ikrom Aulia Fahdi.....	118
Gambar A.11 Lembar Kritik dan Saran Ikrom Aulia Fahdi	119
Gambar A.12 Lembar <i>Usability Testing</i> Ikrom Aulia Fahdi	120
Gambar A.13 Lembar Kuisioner Luthfi Soehadak.....	121
Gambar A.14 Lembar Kritik dan Saran Luthfi Soehadak	122
Gambar A.15 Lembar <i>Usability Testing</i> Luthfi Soehadak	123
Gambar A.16 Lembar Kuisioner Nusantara Jaya Sakti	124
Gambar A.17 Lembar Kritik dan Saran Nusantara Jaya Sakti ..	125
Gambar A.18 Lembar <i>Usability Testing</i> Nusantara Jaya Sakti ..	126
Gambar A.19 Lembar Kuisioner Zakwan Hilmy	127
Gambar A.20 Lembar Kritik dan Saran Zakwan Hilmy.....	128
Gambar A.21 Lembar <i>Usability Testing</i> Zakwan Hilmy.....	129
Gambar A.22 Lembar Kuisioner Dwiky Okka T	130
Gambar A.23 Lembar Kritik dan Saran Dwiky Okka T.....	131
Gambar A.24 Lembar <i>Usability Testing</i> Dwiky Okka T.....	132
Gambar A.25 Lembar Kuisioner Maria Rizqi Yudha S	133
Gambar A.26 Lembar Kritik dan Saran Maria Rizqi Yudha S..	134
Gambar A.27 Lembar <i>Usability Testing</i> Maria Rizqi Yudha S.	135
Gambar A.28 Lembar Kuisioner Lydia Angela	136
Gambar A.29 Lembar Kritik dan Saran Lydia Angela.....	137
Gambar A.30 Lembar <i>Usability Testing</i> Lydia Angela.....	138

DAFTAR TABEL

Tabel 3.1 Spesifikasi Kebutuhan Fungsional	18
Tabel 3.2 Spesifikasi Kasus Penggunaan	21
Tabel 3.3 Skenario Kasus Penggunaan Menelusuri Lingkungan Virtual Mall	22
Tabel 3.4 Skenario Melihat Menu Navigasi.....	23
Tabel 3.5 Memilih Pilihan Navigasi.....	24
Tabel 3.6 Skenario Skenario Memilih Rute Navigasi	25
Tabel 3.7 Skenario Memilih Destinasi	26
Tabel 3.8 Skenario Melihat Rute Terdekat dalam Tampilan Realitas Virtual.....	26
Tabel 3.9 Skenario Melihat Informasi Ruangan.....	27
Tabel 5.1 Pengujian Mencari Menu Navigasi	74
Tabel 5.2 Pengujian Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Sama	75
Tabel 5.3 Pengujian Mengikuti <i>Path</i> yang Terbentuk Untuk Menuju Lokasi Dalam Lantai yang Sama	77
Tabel 5.4 Pengujian Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator	79
Tabel 5.5 Pengujian Mengikuti <i>Path</i> Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Eskalator	81
Tabel 5.6 Pengujian Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift	83
Tabel 5.7 Pengujian Mengikuti <i>Path</i> Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Lift	85
Tabel 5.8 Pengujian Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Sama	87
Tabel 5.9 Pengujian Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator	89

Tabel 5.10 Pengujian Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift91

Tabel 5.11 Rincian *Task* pada *Usability Testing*94

Tabel 5.12 Daftar Pernyataan Kuisisioner95

Tabel 5.13 Rangkuman Hasil Pengujian Fungsionalitas96

Tabel 5.14 Rangkuman Hasil *Usability Testing A*98

Tabel 5.15 Rangkuman Hasil *Usability Testing B*98

Tabel 5.16 Hasil Perbandingan *Usability Testing*99

Tabel 5.17 Rangkuman Hasil Kuisisioner Aspek Kemudahan, Kenyamanan, Immersivity, dan Kegunaan Aplikasi102

Tabel 5.18 Rangkuman Pilihan Jawaban Aspek Kemudahan, Kenyamanan, Immersivity, dan Kegunaan Aplikasi102

DAFTAR KODE SUMBER

Kode Sumber 4.1 Implementasi Kontrol Pergerakan Pengguna	51
Kode Sumber 4.2 Implementasi Informasi Ruangan	53
Kode Sumber 4.3 Fungsi <i>SetNavigationPath</i>	55
Kode Sumber 4.4 Fungsi <i>SetNavigationPathFalse</i>	56
Kode Sumber 4.5 Fungsi <i>ResetPath</i>	56
Kode Sumber 4.6 Fungsi <i>SetDestinationFalse</i>	57
Kode Sumber 4.7 Fungsi <i>SetLiftCaseFalse</i>	58
Kode Sumber 4.8 Fungsi <i>SetLiftCase</i>	58
Kode Sumber 4.9 Fungsi Untuk Mengatur Pilihan Destinasi	59
Kode Sumber 4.10 Implementasi Fitur <i>Navigation(Path)</i>	61
Kode Sumber 4.11 Fungsi <i>CheckPosition</i> Untuk Mengecek Posisi Pengguna	62
Kode Sumber 4.12 Fungsi <i>onTriggerEnter</i> pada <i>Collider</i> Eskalator	63
Kode Sumber 4.13 Implementasi Fungsi Eskalator	63
Kode Sumber 4.14 Implementasi Fitur <i>Navigation(path)</i> pada Lantai yang Berbeda Melalui Lift	66
Kode Sumber 4.15 Kode Untuk Perpindahan Objek Pengguna pada Lift	67
Kode Sumber 4.16 Implementasi Fitur <i>Navigation(Auto)</i>	68
Kode Sumber 4.17 Implementasi Perpindahan Lift pada Fitur <i>Navigation(Auto)</i>	68
Kode Sumber 4.18 Implementasi Skenario Tutorial	71
Kode Sumber 4.19 Implementasi Perpindahan Scene	72

(Halaman ini sengaja dikosongkan)

BAB I

PENDAHULUAN

Bab ini menjelaskan tentang hal-hal yang menjadi latar belakang, permasalahan yang dihadapi, batasan masalah, tujuan dan manfaat, metodologi dan sistematika penulisan yang digunakan dalam pembuatan tugas akhir ini.

1.1 Latar Belakang

Saat ini banyak bangunan besar dan bertingkat di kota-kota besar. Salah satu contoh dari bangunan tersebut adalah mall. Mall adalah jenis dari pusat perbelanjaan yang secara arsitektur berupa bangunan tertutup dengan pengatur suhu, memiliki koridor dengan posisi toko yang saling berhadapan, sehingga nyaman bagi pengunjung untuk jalan-jalan dan berbelanja. Pada umumnya mall terdiri dari dua sampai tiga lantai, dengan banyak ruangan tiap lantai.

Bagi setiap orang yang baru mengunjungi suatu mall sebagian besar akan kesulitan untuk mencari lokasi dari tempat yang ingin dituju pada mall tersebut. Hal ini dikarenakan bangunan mall memiliki area yang cukup luas, bertingkat-tingkat, dan terdiri dari banyak ruang. Dengan arsitektur bangunan yang sedemikian rupa, pastinya orang akan kebingungan apabila disuruh mencari lokasi pada suatu mall, terutama pada mall yang baru saja dikunjungi.

Solusi yang sudah ada untuk mengatasi permasalahan ini adalah dibuatnya penunjuk arah yang diletakkan di beberapa tempat pada mall, papan informasi yang biasanya berisikan denah yang menggambarkan ruangan-ruangan yang ada pada mall. Selain itu pada mall juga disediakan fasilitas *information center* yang berfungsi sebagai pusat informasi yang dapat didatangi oleh pengunjung mall untuk menanyakan hal-hal yang berkaitan dengan mall yang bersangkutan, mulai dari lokasi ruangan, *event* yang sedang berlangsung, dan sebagainya. Pihak mall juga telah menyediakan satpam yang berkeliling dalam mall sebagai pemberi informasi dan penunjuk jalan untuk orang yang kebingungan dalam

mencari lokasi pada mall yang bersangkutan. Selain itu, beberapa mall juga menyediakan *website* sebagai media informasi bagi masyarakat umum tentang mall yang bersangkutan. Namun kenyataannya, tidak semua mall memiliki *website*, dan tidak semua mall yang memiliki *website* mencantumkan denah dari mall yang bersangkutan. Dengan menggunakan solusi diatas, masih ada saja orang yang kebingungan dan tersesat saat berada di mall. Hal tersebut dikarenakan tidak semua orang memanfaatkan solusi yang telah disediakan oleh mall tersebut. Untuk menyelesaikan permasalahan ini, perlu adanya solusi lain yang dapat dijadikan alternatif dari solusi yang ada dengan memanfaatkan teknologi yang ada.

Saat ini sedang berkembang teknologi realitas virtual yang dapat membuat orang merasakan lingkungan dari suatu tempat yang disimulasikan dengan komputer tanpa harus datang ke lingkungan yang sebenarnya. Teknologi ini dapat digunakan untuk memodelkan sebuah bangunan ke dalam sebuah lingkungan virtual, dan bangunan mall dapat diaplikasikan pada teknologi ini. Selain itu, fungsi dari informasi-informasi yang disediakan oleh pihak mall pada dunia nyata seperti penunjuk arah, papan informasi, *information center*, dan satpam dapat dikemas dalam bentuk sistem navigasi dan dapat ditambahkan pada aplikasi realitas virtual.

Untuk dapat menggunakan teknologi realitas virtual, sudah ada perangkat Google Cardboard yang membuat pengguna dapat mengakses lingkungan virtual hanya dengan bermodalkan *smartphone*. Selain itu, harga dari perangkat Google Cardboard sangatlah terjangkau oleh setiap kalangan. Di lain teknologi, Unity memiliki sebuah fitur yang mendukung para developer untuk melakukan navigasi dalam lingkungan 3D, yaitu NavMesh. Dengan melakukan eksplorasi yang lebih dalam tentang fitur-fitur yang ada, NavMesh dapat dikembangkan untuk membuat navigasi dalam lingkungan virtual untuk memudahkan pengguna menuju lokasi yang dituju.

Dengan adanya permasalahan dan teknologi yang sudah berkembang, maka topik dari tugas akhir ini adalah membuat sistem

navigasi pada lingkungan virtual mall dengan mengaplikasikan fitur NavMesh pada Unity3D dan teknologi Google Cardboard. Dengan sistem navigasi tersebut diharapkan lebih memudahkan pengguna aplikasi realitas virtual dalam melakukan pencarian terhadap lokasi yang dituju dalam lingkungan virtual.

1.2 Rumusan Masalah

Rumusan masalah yang diangkat dalam tugas akhir ini adalah sebagai berikut:

1. Bagaimana mengintegrasikan lingkungan virtual yang telah dibuat dengan perangkat Google Cardboard?
2. Bagaimana mengimplementasikan fitur NavMesh pada Unity 3D untuk membuat sistem navigasi pada lingkungan virtual bangunan bertingkat seperti mall?

1.3 Batasan Masalah

Permasalahan yang dibahas pada tugas akhir ini memiliki beberapa batasan, di antaranya sebagai berikut:

1. Bahasa pemrograman yang digunakan adalah Bahasa C#
2. Pembangunan aplikasi dilakukan dengan menggunakan IDE Visual Studio dan Unity 3D serta memanfaatkan teknologi Google Cardboard
3. Sistem navigasi yang dibuat memanfaatkan fitur NavMesh pada Unity 3D
4. Input lokasi yang dituju sudah disediakan dalam aplikasi
5. Studi kasus yang digunakan adalah bangunan mall East Coast Center – Pakuwon, Surabaya sampai dengan lantai 3
6. Aplikasi yang dibangun hanya berjalan pada perangkat bergerak Android

1.4 Tujuan

Tujuan dari pembuatan tugas akhir ini antara lain:

1. Membangun sistem navigasi pada lingkungan virtual mall dengan menggunakan NavMesh dan Google Cardboard

1.5 Manfaat

Manfaat dari hasil pembuatan tugas akhir ini antara lain:

1. Memudahkan pengguna aplikasi realitas virtual untuk mengunjungi lokasi yang ingin dituju yang berada dalam lingkungan virtual
2. Sebagai alat bantu untuk mempresentasikan kepada pengguna tentang bangunan mall terkait.
3. Dapat menjadi referensi bagi penelitian tugas akhir (TA) yang lain.

1.6 Metodologi

Pembuatan Tugas Akhir dilakukan dengan menggunakan metodologi sebagai berikut:

A. Studi literatur

Tahap studi literatur merupakan tahap pembelajaran dan pengumpulan informasi yang digunakan untuk mengimplementasikan tugas akhir. Tahap ini diawali dengan pengumpulan literatur, diskusi, eksplorasi teknologi dan pustaka, serta pemahaman dasar teori yang digunakan pada topik tugas akhir. Literatur-literatur yang dimaksud disebutkan sebagai berikut:

1. Realitas Virtual
2. Google Cardboard
3. Blender
4. Pemodelan Tiga Dimensi (3D)
5. Unity 3D
6. Microsoft Visual Studio
7. Navmesh

8. Line Renderer

B. Analisis dan Perancangan Perangkat Lunak

Pada tahap ini diawali dengan melakukan analisis awal terhadap permasalahan utama yang muncul pada topik tugas akhir. Kemudian dilakukan perancangan perangkat lunak yang meliputi penentuan data yang digunakan dan proses-proses yang akan dilaksanakan. Perancangan yang akan dibuat dibuat antara lain meliputi perancangan skenario kasus penggunaan, perancangan arsitektur, perancangan antarmuka dan kontrol, perancangan lingkungan virtual mall, perancangan objek pengguna, perancangan area navigasi, dan perancangan sistem navigasi.

C. Implementasi dan Pembuatan Sistem

Pada tahap ini akan dilakukan implementasi dari perancangan yang telah dibuat. Implementasi yang dilakukan meliputi pembuatan lingkungan virtual mall, pembuatan area navigasi, pembuatan objek pengguna, dan pembuatan sistem navigasi.

D. Uji coba dan evaluasi

Pada tahap ini dilakukan uji coba dengan menggunakan beberapa macam kondisi untuk mencoba aplikasi dapat berjalan atau tidak. Uji fungsionalitas untuk mengetahui apakah aplikasi sudah memenuhi kebutuhan fungsionalitas. Pengujian akan dilakukan oleh 10 orang pengguna. Pengguna akan diminta menggunakan aplikasi kemudian diminta untuk mengisi lembar kuisioner untuk memberikan umpan baliknya. Kemudian akan dilakukan evaluasi untuk memeriksa ketepatan dari fitur navigasi dalam menunjukkan rute menuju tempat tujuan dan apakah sudah nyaman digunakan oleh pengguna.

E. Penyusunan Buku Tugas Akhir

Pada tahap ini dilakukan penyusunan laporan yang berisi dasar teori, dokumentasi dari perangkat lunak, dan hasil-hasil yang diperoleh selama pengerjaan tugas akhir.

Sistematika penulisan buku tugas akhir ini secara garis besar antara lain :

1. Pendahuluan
2. Tinjauan Pustaka
3. Analisis dan Perancangan
4. Implementasi
5. Pengujian dan Evaluasi
6. Kesimpulan dan Saran
7. Daftar Pustaka

1.7 Sistematika Penulisan

Buku tugas akhir ini terdiri dari beberapa bab, yang dijelaskan sebagai berikut:

BAB I PENDAHULUAN

Bab ini berisi latar belakang masalah, rumusan dan batasan permasalahan, tujuan dan manfaat pembuatan tugas akhir, metodologi yang digunakan, dan sistematika penyusunan tugas akhir.

BAB II TINJAUAN PUSTAKA

Bab ini membahas dasar pembuatan dan beberapa teori penunjang yang berhubungan dengan pokok pembahasan yang mendasari pembuatan Tugas Akhir ini.

BAB III ANALISIS DAN PERANCANGAN

Bab ini membahas analisis dari sistem yang dibuat meliputi analisis permasalahan, deskripsi umum perangkat lunak, spesifikasi kebutuhan, dan identifikasi pengguna. Kemudian membahas rancangan dari sistem yang dibuat meliputi rancangan skenario kasus penggunaan, arsitektur, lingkungan virtual, sistem navigasi, dan antarmuka.

BAB IV IMPLEMENTASI

Bab ini membahas implementasi dari rancangan sistem yang dilakukan pada tahap perancangan. Penjelasan implementasi

meliputi implementasi pembuatan aplikasi realitas virtual sistem navigasi dengan menggunakan Navmesh dan Google Cardboard, dan antarmuka aplikasi.

BAB V PENGUJIAN DAN EVALUASI

Bab ini membahas pengujian dari aplikasi yang dibuat dengan melihat keluaran yang dihasilkan oleh aplikasi dan evaluasi untuk mengetahui kemampuan aplikasi.

BAB VI PENUTUP

Bab ini berisi kesimpulan dari hasil pengujian yang dilakukan serta saran untuk pengembangan aplikasi selanjutnya.

(Halaman ini sengaja dikosongkan)

BAB II

TINJAUAN PUSTAKA

Bab ini berisi penjelasan teori-teori yang berkaitan dengan metode yang diajukan pada pengimplementasian preangkat lunak. Penjelasan ini bertujuan untuk memberikan gambaran umum terhadap sistem yang dibuat dan berguna sebagai penunjang dalam pengembang perangkat lunak.

2.1 Realitas Virtual

Realitas virtual dapat didefinisikan sebagai sebuah teknologi yang membuat pengguna dapat berinteraksi dengan suatu lingkungan yang dikomputasi oleh komputer. Realitas virtual dapat menciptakan sebuah pengalaman sensorik. Kebanyakan lingkungan realitas virtual saat ini memberikan pengalaman visual yang ditampilkan baik pada layar komputer atau pada penampil *stereoskopik* khusus, tetapi beberapa simulasi diberi penambahan informasi sensorik tambahan seperti suara melalui pengeras suara atau sejenisnya[1].

Realitas virtual berbeda dengan animasi maupun video yang citranya dimainkan atau diulangi dalam suatu sekuen yang sudah diatur, realitas virtual bisa dilihat, berinteraksi dan melihat dari berbagai perspektif. Sehingga memberikan fleksibilitas yang lebih besar dari biasanya. Gambar 2.1 adalah contoh penggunaan realitas virtual dalam aplikasi komputer.



Gambar 2.1 Permainan dengan Memanfaatkan Teknologi Realitas Virtual

2.2 Google Cardboard

Google Cardboard adalah perangkat *virtual reality* yang dikembangkan oleh Google dengan bahan karton yang dilipat dan menggunakan handphone sebagai layarnya. Contoh penggunaan Google Cardboard adalah dalam bermain game *Virtual Reality*, maka pengguna seolah-olah berada dalam dunia game tersebut, Contoh penggunaan lainnya adalah untuk menonton video 3D, virtual tour, dan untuk menonton film 360°.

Versi pertama dari Google Cardboard memiliki metode input yang memanfaatkan cincin magnet yang bila ditarik, bisa dirasakan oleh magnetometer headphone dan dibaca sebagai sebuah event masukan. Pada Google Cardboard versi 2, *trigger* input berupa cincin magnet telah diganti dengan tombol masukan universal berbasis sentuhan melalui elektrostatis yang dapat bekerja pada semua *smartphone*. Google Cardboard versi 2 juga dapat digunakan pada *smartphone* yang memiliki ukuran besar, yakni dengan ukuran maksimal layar sampai dengan 6 inch, berbeda dengan versi 1 yang memiliki ukuran maksimal layar sampai dengan 5 inch.

Untuk memudahkan para pengembang aplikasi untuk membuat aplikasi berbasis Google Cardboard, Google telah menyediakan *Software Development Kit* (SDK) untuk mempermudah pekerjaan tersebut yaitu Cardboard SDK. Cardboard SDK berisi berbagai kode sumber yang dapat digunakan oleh pengembang. Selain kode sumber, Cardboard SDK juga diperlukan untuk menyambungkan perangkat Google Cardboard dengan berbagai aplikasi pengembangan aplikasi dan permainan, sebagai contoh Unity 3D. Cardboard SDK juga berisi kode sumber untuk fitur-fitur standar yang ada pada Unity untuk pengembangan permainan, seperti fungsi untuk pengaturan gerak pemain, gerakan kamera, dan fitur-fitur lainnya. Cardboard SDK ini juga dilengkapi dengan dokumentasi yang cukup jelas sehingga pengembang dapat merubah atau mengembangkan SDK ini menjadi sesuai dengan keinginan. Cardboard SDK dapat diunduh melalui situs resmi Unity atau pada forum-forum diskusi mengenai Unity dan Google Cardboard.

2.3 Blender

Blender adalah perangkat lunak grafika komputer 3D yang bersifat *open-source*. Perangkat lunak ini digunakan untuk membuat film animasi, efek visual, model cetak 3D, aplikasi 3D interaktif dan permainan video. Blender memiliki beberapa fitur termasuk pemodelan 3D, penteksturan, penyuntingan gambar bitmap, penulangan, simulasi cairan dan asap, simulasi partikel, animasi, penyunting video, pemahat digital, dan rendering[2].

2.4 Pemodelan Tiga Dimensi (3D)

Tiga dimensi atau bisa disingkat 3D atau yang biasa disebut ruang, adalah bentuk yang memiliki panjang, lebar, dan tinggi. Sehingga bentuk tersebut dapat dilihat dari segala sisi. Istilah biasanya digunakan dalam bidang seni, animasi, komputer, dan matematika.

Pemodelan adalah membentuk suatu benda-benda atau objek. Membuat dan mendesain objek tersebut sehingga terlihat seperti hidup. Sesuai dengan objek dan basisnya, proses ini secara keseluruhan dikerjakan di komputer. Melalui konsep dan proses desain, keseluruhan objek bisa diperlihatkan secara tiga dimensi, sehingga banyak yang menyebut hasil ini sebagai pemodelan tiga dimensi[3].

2.5 Unity 3D

Unity atau Unity 3D adalah sebuah perangkat lunak yang berfungsi untuk membangun permainan atau aplikasi. Unity merupakan suatu *game development ecosystem* yang mampu digunakan untuk membuat permainan atau aplikasi dalam berbagai macam *platform* baik *console*, *desktop*, dan *mobile*. Bahasa pemrograman utama Unity adalah C# dengan IDE Mono Develop [4].

Fitur *scripting* yang disediakan, mendukung 3 bahasa pemrograman, JavaScript, C#, dan Boo. *Flexible and EasyMoving*, *rotating*, dan *scaling objects* hanya perlu sebaris kode. Begitu juga dengan *Duplicating*, *removing*, dan *changing properties*. *Visual*

Properties Variables yang didefinisikan dengan scripts ditampilkan pada *Editor*. Bisa digeser, di *drag* and *drop*, bisa memilih warna dengan *color picker*. Berbasis .NET. Artinya penjalanan program dilakukan dengan *Open Source* .NET platform, Mono serta mendukung pengembangan aplikasi Microsoft, SONY, Qualcomm, BlackBerry, Samsung, Nintendo, Oculus VR dan intel.

2.6 Microsoft Visual Studio

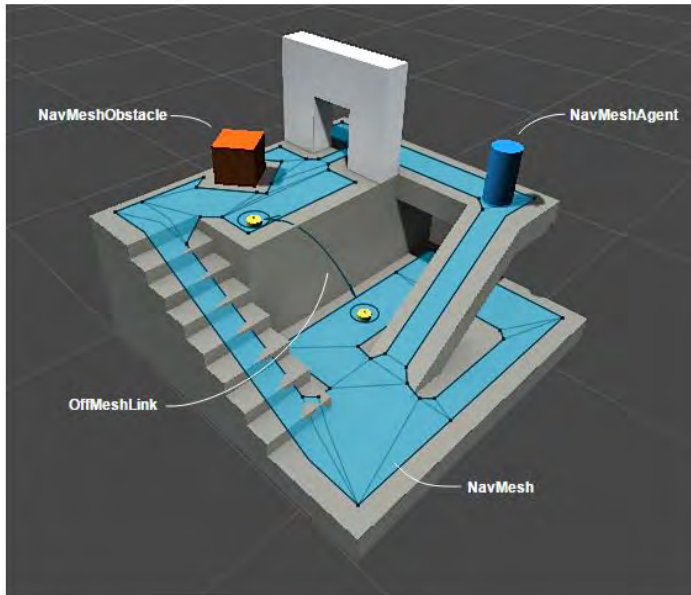
Microsoft Visual Studio merupakan sebuah aplikasi yang terdiri dari kompiler, SDK, *Integrated Development Environment (IDE)*, dan dokumentasi yang biasa digunakan untuk mengembangkan aplikasi personal, aplikasi *web*, aplikasi *mobile*, aplikasi konsol maupun komponen aplikasi pada Microsoft Visual Studio. Banyak compiler yang disediakan oleh Microsoft Visual Studio, misalnya saja Visual C++, Visual C#, Visual Basic, Visual Basic .NET, Visual InterDev, Visual J++, Visual J#, Visual FoxPro, dan Visual SourceSafe. Banyak pengembang aplikasi yang mengembangkan aplikasinya menggunakan Microsoft Visual Studio karena banyak fungsi yang telah disediakan oleh Microsoft Visual Studio, sehingga pengembang lebih mudah dan sangat terbantu dalam mengembangkan maupun membuat aplikasi mereka [5].

2.7 Navmesh

Navmesh adalah sebuah class pada *library* UnityEngine yang dapat digunakan untuk melakukan spatial query, seperti *pathfinding* dan *walkability* test, menentukan *pathfinding cost* dalam lingkup area tertentu, dan untuk melakukan adaptasi global behavior dari *pathfinding* dan rintangan yang ada pada suatu area [6].

Navigation Mesh merupakan suatu bentuk struktur data yang terutama sangat dibutuhkan dalam *pathfinding* dalam lingkungan 3D. *Mesh* merupakan suatu bentuk yang dapat dianggap sebagai graf dengan node yang berbentuk polygon. Dalam *pathfinding*, setiap bentuk polygon dalam *navigation mesh* akan dijadikan node dari graf, dengan nilai koordinat biasanya diambil dari titik tengah polygon tersebut. Dalam melakukan *pathfinding*, Navmesh

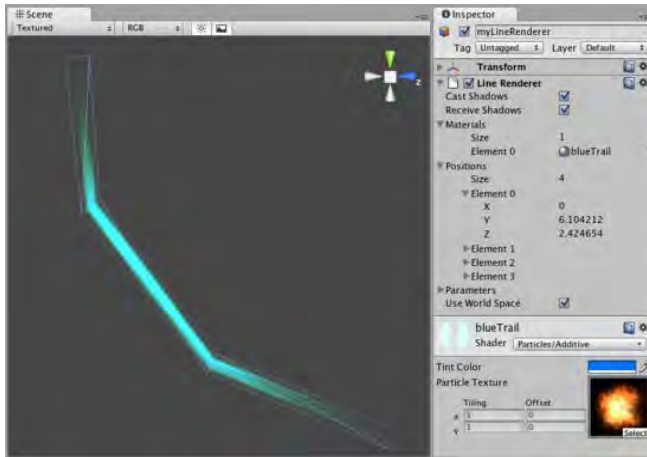
menggunakan algoritma A* untuk mencari rute yang paling optimal[7]. Gambaran dari implementasi Navmesh pada Unity dapat dilihat pada Gambar 2.2.



Gambar 2.2 Gambaran Implementasi Navmesh pada Unity

2.8 Line Renderer

Line Renderer adalah sebuah *class* dalam Unity yang dapat digunakan untuk membuat garis lurus antara dua titik atau lebih dalam lingkungan 3D. Sebuah komponen Line Renderer dapat digunakan untuk membuat garis mulai dari garis lurus yang sederhana, sampai garis spiral yang kompleks. Garis yang dibentuk oleh satu komponen Line Renderer terus berkelanjutan. Apabila pengguna ingin menambahkan garis lain secara terpisah, pengguna harus menggunakan beberapa *GameObject* lain, dan masing-masing *GameObject* memiliki komponen Line Renderer. Contoh pemngimplementasian Line Renderer pada Unity dapat dilihat pada Gambar 2.3.



Gambar 2.3 Implementasi Line Renderer pada Unity

BAB III

ANALISIS DAN PERANCANGAN

Bab ini menjelaskan tentang analisis dan perancangan aplikasi realitas virtual sistem navigasi pada lingkungan virtual mall dengan menggunakan NavMesh dan Google Cardboard. Pembahasan yang akan dilakukan meliputi analisis fitur yang dibutuhkan dan perancangan perangkat lunak.

3.1 Analisis

Tahap analisis dibagi menjadi beberapa bagian antara lain analisis permasalahan, analisis kebutuhan, analisis pengguna, dan deskripsi umum aplikasi.

3.1.1 Analisis Permasalahan

Permasalahan yang diangkat pada tugas akhir ini adalah sulitnya orang yang baru mengunjungi suatu mall untuk mencari lokasi-lokasi yang ada pada mall tersebut. Hal ini dikarenakan bangunan mall memiliki area yang cukup luas, bertingkat-tingkat, dan terdiri dari banyak ruang. Dan terkadang seseorang dapat tersesat dalam bangunan tersebut dikarenakan informasi yang disediakan pada mall jarang digunakan.

Solusi yang sudah ada untuk mengatasi permasalahan ini adalah dibuatnya penunjuk arah yang diletakkan di beberapa tempat pada mall, papan informasi yang biasanya berisikan denah yang menggambarkan ruangan-ruangan yang ada pada mall. Selain itu pada mall juga disediakan fasilitas *information center* yang berfungsi sebagai pusat informasi yang dapat didatangi oleh pengunjung mall untuk menanyakan hal-hal yang berkaitan dengan mall yang bersangkutan, mulai dari lokasi ruangan, *event* yang sedang berlangsung, dan sebagainya. Pihak mall juga telah menyediakan satpam yang berkeliling dalam mall sebagai pemberi informasi dan penunjuk jalan untuk orang yang kebingungan dalam mencari lokasi pada mall yang bersangkutan. Selain itu, beberapa mall juga menyediakan *website* sebagai media informasi

bagi masyarakat umum tentang mall yang bersangkutan. Namun kenyataannya, tidak semua mall memiliki *website*, dan tidak semua mall yang memiliki *website* mencantumkan denah dari mall yang bersangkutan. Dengan menggunakan solusi diatas, masih ada saja orang yang kebingungan dan tersesat saat berada di mall. Hal tersebut dikarenakan tidak semua orang memanfaatkan solusi yang telah disediakan oleh mall tersebut. Untuk menyelesaikan permasalahan ini, perlu adanya solusi lain yang dapat dijadikan alternatif dari solusi yang ada dengan memanfaatkan teknologi yang ada.

Saat ini sedang berkembang teknologi realitas virtual yang dapat membuat orang merasakan lingkungan dari suatu tempat yang disimulasikan dengan komputer tanpa harus datang ke lingkungan yang sebenarnya. Teknologi ini dapat digunakan untuk memodelkan sebuah bangunan ke dalam sebuah lingkungan virtual, yang dalam studi kasus ini bangunan yang akan dimodelkan adalah bangunan mall. Selain itu, fungsi dari informasi-informasi yang disediakan oleh pihak mall pada dunia nyata seperti penunjuk arah, papan informasi, *information center*, dan satpam dapat dikemas dalam bentuk sistem navigasi dan dapat disisipkan pada aplikasi realitas virtual. Untuk membuat sistem navigasi, Unity telah menyediakan fitur untuk memudahkan developer dalam membuat sistem navigasi, yaitu NavMesh. Terlebih, Google memiliki perangkat yang dapat mengakses teknologi realitas virtual dengan harga yang terjangkau oleh semua kalangan, yaitu Google Cardboard. Teknologi ini dapat digunakan sebagai solusi untuk permasalahan diatas.

Aplikasi yang akan dibuat merupakan sistem navigasi berbasis realitas virtual pada lingkungan virtual mall yang akan memanfaatkan NavMesh pada Unity dan Google Cardboard. Aplikasi ini akan memodelkan sebuah mall kedalam lingkungan virtual, dengan diberi suatu sistem navigasi didalamnya. Sistem navigasi yang dibuat akan membantu pengguna aplikasi dalam melakukan pencarian rute terdekat dalam lingkungan virtual yang disediakan.

Diharapkan aplikasi ini dapat memudahkan pengguna aplikasi realitas virtual dalam melakukan pencarian terhadap lokasi yang dituju dalam lingkungan virtual.

3.1.2 Deskripsi Umum Aplikasi

Aplikasi yang akan dibuat dalam tugas akhir ini adalah sebuah aplikasi realitas virtual sistem navigasi pada lingkungan virtual mall dengan memanfaatkan NavMesh pada Unity dan Google Cardboard. Aplikasi ini akan memvisualisasikan objek 3D yang sudah diatur membentuk bangunan mall hingga membentuk lingkungan virtual. Pada tugas akhir ini, bangunan mall yang dimodelkan adalah bangunan East Coast Center – Pakuwon. Aplikasi ini akan menampilkan lingkungan virtual yang dibentuk pada *smartphone*, dan dapat dilihat dengan tampilan realitas virtual melalui perangkat Google Cardboard. Dengan perangkat Google Cardboard, pengguna akan merasakan sensasi seolah-olah berada di area mall dalam lingkungan virtual.

Fitur yang dimiliki aplikasi ini secara garis besar terdiri dari dua fitur, yaitu “Navigation (path)” dan “Navigation (auto)”. Pada fitur “Navigation (path)”, pengguna dapat mencari rute terdekat dari posisi pengguna sekarang ke destinasi yang dipilih dengan menggambarkan garis menghubungkan antara posisi pengguna dengan destinasi. Apabila destinasi berbeda lantai dengan posisi pengguna sekarang, pengguna dapat memilih rute yang harus dilewati untuk menuju destinasi tersebut, yaitu melalui eskalator atau lift. Pada fitur “Navigation (auto)”, pengguna dapat mencari rute terdekat dari posisi pengguna sekarang ke destinasi yang dipilih dengan cara digerakkan secara otomatis oleh sistem menuju destinasi. Sama dengan fitur sebelumnya, pengguna juga dapat memilih rute yang harus dilewati apabila destinasi berbeda lantai, yaitu melalui eskalator atau lift.

3.1.3 Analisis Kebutuhan

Pada tahap ini akan dijelaskan kebutuhan yang dibutuhkan pengguna dalam menggunakan aplikasi. Tahap analisis kebutuhan

dibagi menjadi dua bagian, antara lain analisis kebutuhan fungsional dan analisis kebutuhan non-fungsional.

3.1.3.1 Analisis Kebutuhan Fungsional

Kebutuhan utama dalam aplikasi ini adalah pengguna dapat menelusuri lingkungan virtual mall yang telah dibuat dan pengguna dapat mencari rute terdekat menuju ruangan yang diinginkan dalam lingkungan virtual menggunakan sistem navigasi secara realitas virtual. Rincian kebutuhan fungsional dapat dilihat pada Tabel 3.1.

Tabel 3.1 Spesifikasi Kebutuhan Fungsional

Kode Kebutuhan	Kebutuhan Fungsional	Deskripsi
F-0001	Menelusuri Lingkungan Mall	Pengguna dapat menelusuri lingkungan mall yang ada dalam aplikasi
F-0002	Melihat menu navigasi	Pengguna dapat melihat menu navigasi yang terdapat di aplikasi
F-0003	Menggunakan fitur navigasi	Pengguna dapat menggunakan fitur navigasi sesuai dengan pilihan jenis navigasi yang disediakan aplikasi
F-0004	Melewati eskalator	Pengguna dapat melewati eskalator untuk menuju destinasi yang berbeda lantai dengan posisi pengguna
F-0005	Melewati lift	Pengguna dapat melewati lift untuk menuju destinasi yang berbeda lantai dengan posisi pengguna

F-0006	Memilih destinasi	Pengguna dapat memilih destinasi sesuai destinasi yang diinginkan pengguna
F-0007	Melihat rute terdekat	Pengguna dapat melihat rute terdekat yang ditampilkan oleh sistem navigasi untuk menuju ke destinasi
F-0008	Melihat informasi ruangan	Pengguna dapat melihat informasi ruangan tempat pengguna berada

3.1.3.2 Analisis Kebutuhan Non-Fungsional

Pada aplikasi ini, terdapat beberapa kebutuhan non-fungsional yang mendukung jalannya aplikasi. Rincian kebutuhan non-fungsional dalam aplikasi ini adalah sebagai berikut :

- a. Kemudahan dalam penggunaan aplikasi
Controller dalam aplikasi ini hanya menggunakan satu perangkat yaitu Google Cardboard. Magnet pada Google Cardboard digunakan untuk memicu sebuah *event* pada aplikasi, seperti berjalan, memencet suatu tombol, dan melakukan navigasi. Untuk melihat lingkungan sekitar dalam lingkungan virtual, tinggal menggerakkan Google Cardboard ke arah pandangan yang diinginkan maka tampilan akan berubah sesuai dengan pandangan yang diinginkan. Hal ini dapat memudahkan pengguna karena perangkat yang dibutuhkan untuk menggunakan aplikasi hanya sebuah *smartphone* dan Google Cardboard.
- b. *Immersivity* dari lingkungan virtual
Immersivity adalah sebuah kondisi dimana pengguna realitas virtual merasakan sensasi sebagaimana saat berada di dunia nyata. Pengguna aplikasi merasa seperti berada di lingkungan mall saat menggunakan aplikasi. Tingkat *immersivity* mempengaruhi tingkat kemiripan objek tiga

dimensi yang dibuat dengan objek yang ada pada dunia nyata.

3.1.4 Analisis Pengguna

Dalam aplikasi ini hanya memiliki satu aktor, yaitu orang yang menggunakan aplikasi realitas virtual sistem navigasi pada lingkungan virtual mall dengan menggunakan NavMesh dan Google Cardboard.

3.2 Perancangan Aplikasi

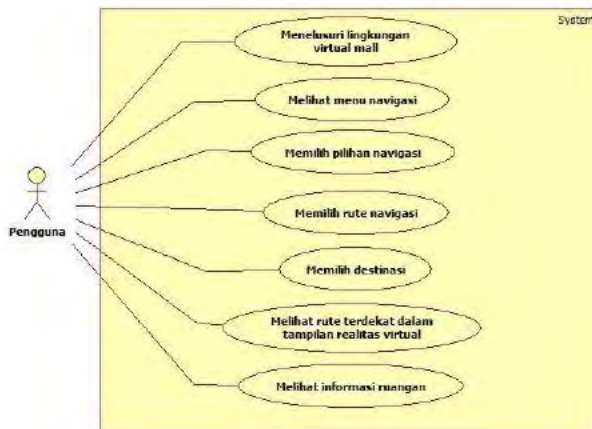
Tahap perancangan dalam subbab ini dibagi menjadi beberapa bagian, yaitu perancangan diagram kasus penggunaan, perancangan arsitektur aplikasi, perancangan lingkungan virtual mall, perancangan area navigasi, perancangan objek pengguna, perancangan informasi pengguna, dan perancangan sistem navigasi. Selain itu terdapat juga perancangan skenario tutorial sebagai gambaran awal kepada pengguna saat menggunakan aplikasi.

3.2.1 Perancangan Skenario Kasus Penggunaan

Berdasarkan tahap analisis yang telah dilakukan, mulai dari analisis permasalahan, analisis kebutuhan, sampai analisis pengguna maka dirancanglah skenario kasus penggunaan aplikasi. Perancangan skenario kasus penggunaan dibagi menjadi dua bagian, yaitu pembuatan diagram kasus penggunaan, kemudian pendefinisian tiap kasus penggunaan.

3.2.1.1 Diagram Kasus Penggunaan

Dalam aplikasi yang dibuat pada tugas akhir ini, terdapat tujuh kasus penggunaan yang ada yaitu menelusuri lingkungan virtual mall, melihat menu navigasi, memilih jenis navigasi, memilih rute navigasi, memilih destinasi, melihat rute terdekat dalam tampilan realitas virtual, dan melihat informasi ruangan. Diagram kasus penggunaan dapat dilihat pada Gambar 3.1.



Gambar 3.1 Diagram Kasus Penggunaan

Penjelasan singkat dari masing-masing kasus penggunaan dapat dilihat pada Tabel 3.2.

Tabel 3.2 Spesifikasi Kasus Penggunaan

Kode Kebutuhan	Nama Kasus Penggunaan	Deskripsi
UC-0001	Menelusuri Lingkungan Virtual Mall	Pengguna dapat menelusuri lingkungan virtual mall yang ada dalam aplikasi
UC-0002	Melihat menu navigasi	Pengguna dapat melihat menu navigasi yang terdapat di aplikasi
UC-0003	Memilih pilihan navigasi	Pengguna dapat memilih pilihan navigasi sesuai dengan pilihan yang diinginkan pengguna
UC-0004	Memilih rute navigasi	Pengguna dapat memilih pilihan rute yang harus dilewati sistem navigasi

		sesuai dengan pilihan yang diinginkan pengguna. Pilihan rute yang disediakan yaitu eskalator dan lift.
UC-0005	Memilih destinasi	Pengguna dapat memilih destinasi sesuai destinasi yang diinginkan pengguna
UC-0006	Melihat rute terdekat dalam tampilan realitas virtual	Pengguna dapat melihat rute terdekat yang ditampilkan oleh sistem navigasi dalam tampilan realitas virtual
UC-0007	Melihat informasi ruangan	Pengguna dapat melihat informasi ruangan tempat pengguna berada

3.2.1.2 Definisi Kasus Penggunaan

Detail mengenai kasus penggunaan tersebut dapat dilihat pada subbab berikut ini.

3.2.1.2.1 Menelusuri Lingkungan Virtual Mall

Skenario kasus penggunaan menelusuri lingkungan virtual mall dapat dilihat pada Tabel 3.3.

Tabel 3.3 Skenario Kasus Penggunaan Menelusuri Lingkungan Virtual Mall

Nama Kasus Penggunaan	Menelusuri Lingkungan Virtual Mall
Kode	UC-0001
Deskripsi	Pengguna dapat menelusuri lingkungan mall yang ada dalam aplikasi
Aktor	Pengguna
Kondisi Awal	Pengguna sudah masuk ke aplikasi, objek sudah di-render, dan <i>smartphone</i> telah terpasang pada

	Google Cardboard
Alur Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna menarik magnet pada Google Cardboard 2. Sistem menerima inputan pengguna dan menggerakkan objek pengguna ke arah depan 3. Pengguna menggerakkan kepala ke sudut tertentu 4. Sistem mengarahkan pandangan pengguna sesuai pergerakan kepala 5. Pengguna menarik magnet pada Google Cardboard 6. Sistem menerima inputan pengguna dan objek pengguna kembali diam 7. Kembali ke alur kejadian normal nomor 1
Alur Kejadian Alternatif	-

3.2.1.2.2 Melihat Menu Navigasi

Skenario kasus penggunaan melihat menu navigasi dapat dilihat pada Tabel 3.4.

Tabel 3.4 Skenario Melihat Menu Navigasi

Nama Kasus Penggunaan	Melihat Menu Navigasi
Kode	UC-0002
Deskripsi	Pengguna dapat melihat menu navigasi yang terdapat pada aplikasi
Aktor	Pengguna
Kondisi Awal	Pengguna sudah masuk ke aplikasi, objek sudah di-render, dan <i>smartphone</i> telah terpasang pada Google Cardboard
Alur Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna menggerakkan kepala ke arah bawah

	2. Pengguna melihat menu navigasi
Alur Kejadian Alternatif	-

3.2.1.2.3 Memilih Pilihan Navigasi

Skenario kasus penggunaan memilih pilihan navigasi dapat dilihat pada Tabel 3.5.

Tabel 3.5 Memilih Pilihan Navigasi

Nama Kasus Penggunaan	Memilih Pilihan Navigasi
Kode	UC-0003
Deskripsi	Pengguna dapat memilih pilihan navigasi sesuai dengan pilihan yang diinginkan pengguna
Aktor	Pengguna
Kondisi Awal	Pengguna sudah melihat menu navigasi
Alur Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih tombol “Navigation (path)” <ol style="list-style-type: none"> A1. Pengguna memilih tombol “Navigation (auto)” A2. Pengguna memilih tombol “Clear” 2. Sistem menerima inputan pengguna dan mengarahkan pengguna menuju menu selanjutnya
Alur Kejadian Alternatif	<ol style="list-style-type: none"> A1. Pengguna memilih tombol “Navigation (auto)” <ol style="list-style-type: none"> 1. Sistem menerima inputan pengguna dan mengarahkan pengguna menuju menu selanjutnya. A2. Pengguna memilih tombol “Clear” <ol style="list-style-type: none"> 1. Sistem menerima inputan pengguna dan mengarahkan pengguna menuju

	menu selanjutnya.
--	-------------------

3.2.1.2.4 Memilih Rute Navigasi

Skenario kasus penggunaan memilih rute navigasi dapat dilihat pada Tabel 3.6.

Tabel 3.6 Skenario Skenario Memilih Rute Navigasi

Nama Kasus Penggunaan	Memilih Rute Navigasi
Kode	UC-0004
Deskripsi	Pengguna dapat memilih pilihan rute yang harus dilewati sistem navigasi sesuai dengan pilihan yang diinginkan pengguna
Aktor	Pengguna
Kondisi Awal	Pengguna sudah memilih pilihan navigasi
Alur Kejadian Normal	<ol style="list-style-type: none"> Pengguna memilih tombol “Eskalator” <ol style="list-style-type: none"> Pengguna memilih tombol “Lift” Pengguna memilih tombol “Back” Sistem menerima inputan pengguna dan mengarahkan pengguna menuju menu selanjutnya
Alur Kejadian Alternatif	<p>A1. Pengguna memilih tombol “Lift”</p> <ol style="list-style-type: none"> Sistem menerima inputan pengguna dan mengarahkan pengguna menuju menu selanjutnya. <p>A2. Pengguna memilih tombol “back”</p> <ol style="list-style-type: none"> Sistem menerima inputan pengguna dan mengarahkan pengguna menuju menu selanjutnya.

3.2.1.2.5 Memilih Destinasi

Skenario kasus penggunaan memilih destinasi dapat dilihat pada Tabel 3.7.

Tabel 3.7 Skenario Memilih Destinasi

Nama Kasus Penggunaan	Memilih Destinasi
Kode	UC-0005
Deskripsi	Pengguna dapat memilih destinasi sesuai dengan pilihan yang diinginkan pengguna
Aktor	Pengguna
Kondisi Awal	Pengguna sudah memilih pilihan rute navigasi
Alur Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna memilih destinasi yang diinginkan 2. Sistem menerima inputan pengguna dan menampilkan rute terdekat sesuai dengan pilihan destinasi pengguna berdasarkan dengan pilihan dan rute navigasi yang dipilih
Alur Kejadian Alternatif	-

3.2.1.2.6 Melihat Rute Terdekat dalam Tampilan Realitas Virtual

Skenario kasus penggunaan melihat rute terdekat dalam tampilan realitas virtual dapat dilihat pada Tabel 3.8.

Tabel 3.8 Skenario Melihat Rute Terdekat dalam Tampilan Realitas Virtual

Nama Kasus Penggunaan	Melihat Rute Terdekat dalam Tampilan Realitas Virtual
Kode	UC-0006
Deskripsi	Pengguna dapat melihat rute terdekat yang ditampilkan oleh sistem navigasi dalam tampilan realitas virtual

Aktor	Pengguna
Kondisi Awal	Pengguna sudah memilih destinasi
Alur Kejadian Normal	<ol style="list-style-type: none"> 1. Sistem menampilkan garis dari lokasi pengguna ke destinasi yang menggambarkan rute terdekat menuju ke destinasi 2. Pengguna mengikuti garis yang ditampilkan oleh sistem
Alur Kejadian Alternatif	-

3.2.1.2.7 Melihat Informasi Ruangan

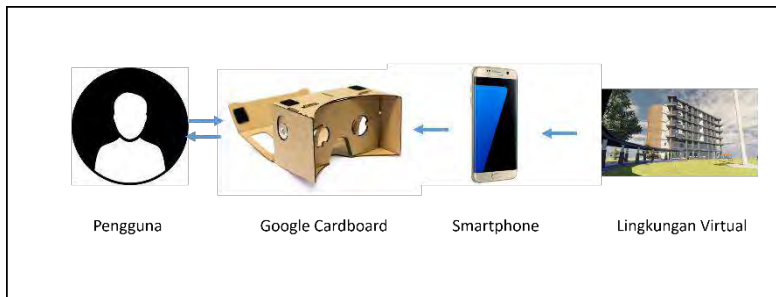
Skenario kasus penggunaan melihat informasi ruangan dapat dilihat pada Tabel 3.9.

Tabel 3.9 Skenario Melihat Informasi Ruangan

Nama Kasus Penggunaan	Melihat Informasi Ruangan
Kode	UC-0007
Deskripsi	Pengguna dapat melihat informasi ruangan berupa nama tempat pengguna berada
Aktor	Pengguna
Kondisi Awal	Pengguna sudah masuk ke aplikasi, objek sudah di-render, dan <i>smartphone</i> telah terpasang pada Google Cardboard
Alur Kejadian Normal	<ol style="list-style-type: none"> 1. Pengguna sampai di suatu ruangan dan berhenti pada ruangan tersebut 2. Sistem menampilkan informasi ruangan selama 5 detik di layar pengguna 3. Pengguna melihat informasi ruangan yang ditampilkan oleh sistem
Alur Kejadian Alternatif	-

3.2.2 Perancangan Arsitektur Aplikasi

Arsitektur sistem pada aplikasi ini didukung oleh beberapa perangkat yaitu *smartphone* dan Google Cardboard. Implementasi aplikasi juga memanfaatkan salah satu *game engine* yang sudah terkenal keunggulannya yaitu Unity. Untuk pembuatan model 3 dimensi menggunakan aplikasi 3D *modelling* Blender. Perancangan arsitektur secara umum aplikasi ini terlihat pada Gambar 3.2.



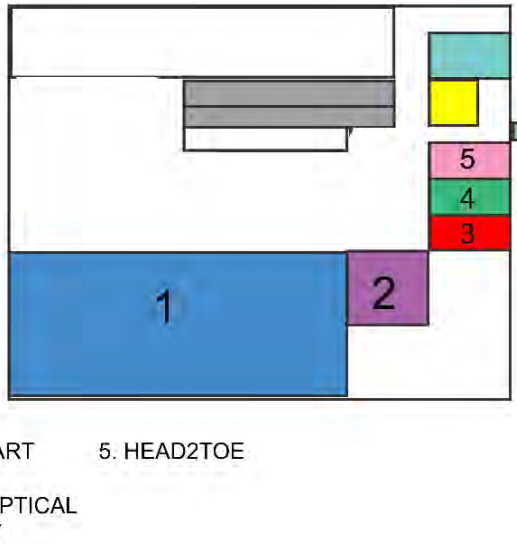
Gambar 3.2 Rancangan Sederhana Arsitektur Aplikasi

3.2.3 Perancangan Lingkungan Virtual Mall

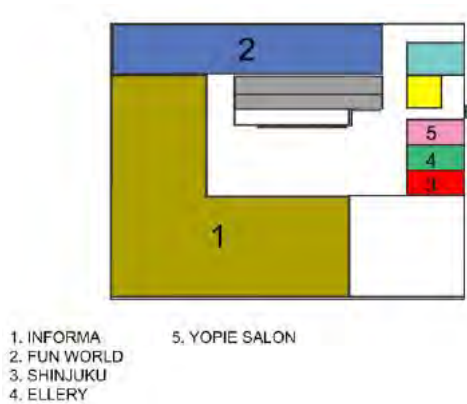
Perancangan lingkungan virtual mall pada aplikasi ini berdasarkan pada denah mall yang ada pada dunia nyata. Denah diperoleh dengan menuju langsung ke mall yang bersangkutan, yaitu gedung East Coast Center – Pakuwon, Surabaya. Lingkungan virtual mall yang akan diimplementasikan adalah gedung East Coast Center – Pakuwon mulai dari lantai 1 sampai dengan lantai 3. Perancangan model ruangan dibuat dengan menggunakan Blender 2.7.6 dengan memanfaatkan add-on Archimesh pada Blender, yang kemudian diimpor kedalam IDE Unity yang nantinya akan diproses sehingga membentuk lingkungan virtual. Gambaran denah yang telah dibuat untuk melakukan perancangan model dapat dilihat pada Gambar 3.3, Gambar 3.4, dan Gambar 3.5.



Gambar 3.3 Denah East Coast Center – Pakuwon Lantai 1



Gambar 3.4 Denah East Coast Center – Pakuwon Lantai 2



Gambar 3.5 Denah East Coast Center – Pakuwon Lantai 3

3.2.4 Perancangan Area Navigasi

Perancangan area navigasi pada aplikasi dibuat dengan menggunakan fitur *Navigation* pada Unity. Dengan menggunakan fitur *Navigation* pada Unity, *navigation mesh* dapat dibuat dan area yang dapat dilewati oleh pengguna dapat dibatasi. Pada fitur *Navigation*, objek yang dapat dilewati oleh pengguna dapat diatur, begitu juga dengan objek yang tidak dapat dilewati pengguna.

3.2.5 Perancangan Objek Pengguna

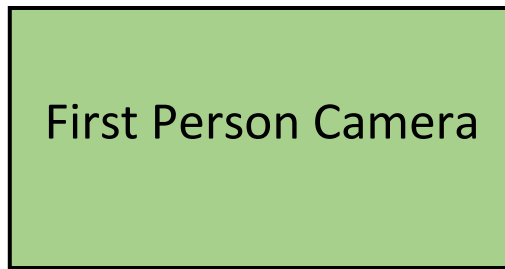
Pada aplikasi ini, pengguna digambarkan sebagai objek *capsule* dalam lingkungan virtual. Kemudian akan dilakukan integrasi antara perangkat Google Cardboard dengan Unity sebagai antarmuka pengguna, perancangan kontrol pengguna saat menggunakan aplikasi, serta pendefinisian objek pengguna sebagai NavMesh Agent.

3.2.5.1 Antarmuka Objek Pengguna

Untuk rancangan tampilan antarmuka objek pengguna, aplikasi ini menerapkan mode *First Person Camera*. Tampilan ini diperoleh dengan melakukan integrasi antara perangkat Google Cardboard dengan Unity menggunakan Cardboard SDK. Objek

pengguna yang telah dibentuk akan diberi kamera dari Cardboard SDK agar pengguna dapat melihat sekeliling dalam lingkungan virtual.

Rancangan antarmuka pengguna dari aplikasi ini hanya berjumlah satu buah layar kosong yang akan diisi oleh objek-objek. Objek yang terlihat pada layar sesuai dengan pergerakan posisi pengguna dalam lingkungan virtual. Rancangan antarmuka pengguna akan terlihat pada Gambar 3.6.



Gambar 3.6 Rancangan Antarmuka Pengguna

3.2.5.2 Kontrol Objek Pengguna

Kontrol objek pengguna pada aplikasi dapat diimplementasikan dengan memberi script pada objek pengguna. Kontrol pengguna pada aplikasi ini adalah dengan memanfaatkan *trigger* pada Google Cardboard dan *pointer* pada aplikasi. Ketika *trigger* ditekan, maka akan memicu suatu *event* dalam aplikasi. Rotasi kamera atau kepala sudah diatur oleh Cardboard SDK. Apabila pengguna menggerakkan kepala ke arah kiri maka pandangan kamera mengarah ke kiri. Apabila pengguna menggerakkan kepala ke arah kanan maka pandangan kamera mengarah ke kanan. Untuk pergerakan pengguna dalam lingkungan virtual juga menggunakan *trigger* pada Google Cardboard. Apabila posisi awal objek pengguna diam kemudian *trigger* ditekan, maka objek pengguna akan berjalan secara otomatis. Sebaliknya apabila objek pengguna sedang dalam kondisi berjalan kemudian *trigger* ditekan, maka objek pengguna

akan berhenti. Untuk pemilihan menu juga menggunakan *trigger* pada Google Cardboard. Pengguna diminta untuk menggerakkan *pointer* pada *button* yang tersedia

3.2.5.3 Pendefinisian Objek Sebagai NavMesh Agent

Pendefinisian objek sebagai NavMesh Agent dilakukan agar objek pengguna yang telah dibuat dapat bergerak sesuai dengan area navigasi yang telah dibentuk. Pergerakan objek pengguna tidak dapat melebihi batas area navigasi yang telah dibuat. Selain itu, pendefinisian objek sebagai NavMesh Agent dibutuhkan agar objek pengguna dapat menggunakan fungsi-fungsi yang terdapat pada NavMesh.

3.2.6 Perancangan Informasi Ruangan

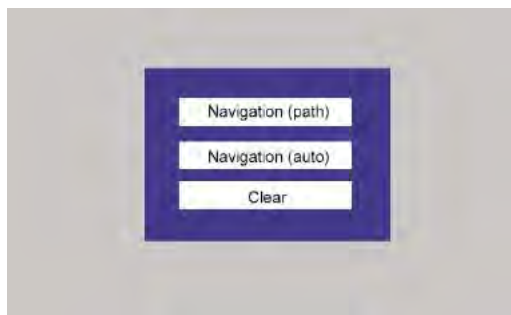
Tampilan informasi ruangan memanfaatkan panel pada Unity dimana panel tersebut akan tampil ketika pengguna berada di ruangan. Tiap ruangan pada lingkungan virtual diberi *collider* dan *trigger* tiap *collider* tersebut diaktifkan. Ketika objek *capsule* yang menggambarkan pengguna bertumbukan dan berada di area *collider* tersebut, maka sistem akan mengaktifkan *trigger* untuk menampilkan informasi ruangan. Informasi ruangan berupa nama dari ruangan tersebut. Ketika pengguna sudah keluar dari area *collider* ruangan, maka sistem akan mengaktifkan *trigger* untuk menonaktifkan informasi ruangan.

3.2.7 Perancangan Sistem Navigasi

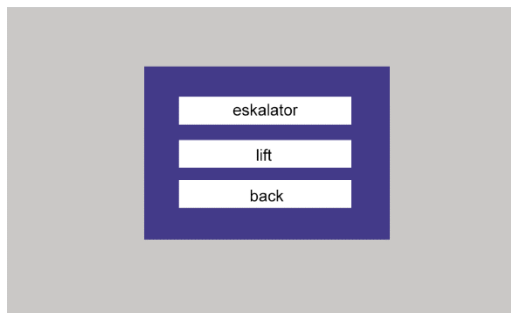
Sistem navigasi pada aplikasi ini memiliki dua buah fitur, yaitu fitur “Navigation (path)” dan “Navigation (auto)”. Fitur “Navigation (path)” digunakan untuk menampilkan navigasi dalam bentuk garis menuju ke destinasi yang dituju. Sedangkan fitur “Navigation (auto)” digunakan untuk melakukan navigasi dengan digerakkan langsung oleh sistem menuju destinasi yang dituju. Untuk melakukan navigasi, pengguna diharuskan menggerakkan kepala ke arah bawah untuk dapat melihat menu navigasi. Terdapat tiga buah tombol dalam menu navigasi, yaitu

tombol “Navigation (path)”, “Navigation (auto)” dan tombol “Clear”. Tombol “Clear” digunakan untuk menghapus rute yang telah terbentuk.

Setelah memilih salah satu tombol antara “Navigation (path)” atau “Navigation (auto)”, maka pengguna akan diarahkan menuju tampilan selanjutnya yang berisi tombol pilihan rute navigasi, yaitu “eskalator” atau “lift”. Pengguna dapat memilih rute navigasi antara melewati eskalator atau lift dengan memilih salah satu tombol tersebut. Pengguna juga dapat memilih tombol “back” untuk dapat kembali ke tampilan menu navigasi. Rancangan antarmuka menu pilihan navigasi dapat terlihat pada Gambar 3.7 dan tampilan pilihan rute navigasi dapat terlihat pada Gambar 3.8.



Gambar 3.7 Rancangan Antarmuka Menu Pilihan Navigasi



Gambar 3.8 Rancangan Antarmuka Pilihan Rute Navigasi

Setelah rute navigasi dipilih, maka akan muncul sebuah panel yang berisikan nama-nama ruangan yang ada pada mall di depan pengguna. Kemudian pengguna diharuskan memilih salah satu ruangan yang ada untuk dijadikan destinasi. Setelah destinasi terpilih, maka sistem navigasi akan menampilkan rute yang harus dilalui pengguna untuk menuju ke destinasi. Apabila saat berada di menu pilihan navigasi pengguna memilih menggunakan fitur “Navigation (path)”, maka sistem navigasi akan menampilkan rute berupa *virtual path* untuk menunjukkan tampilan rute dari posisi pengguna berada menuju destinasi yang dipilih. *Virtual path* dibentuk berdasarkan pencarian jarak terdekat antara posisi pengguna aplikasi dengan destinasi. Dan apabila saat berada di menu pilihan navigasi pengguna memilih menggunakan fitur “Navigation (auto)”, maka sistem akan melakukan navigasi dengan cara menggerakkan pengguna menuju destinasi yang dipilih. Rancangan antarmuka panel pilihan destinasi akan terlihat pada Gambar 3.9, dan rancangan tampilan *virtual path* akan terlihat pada Gambar 3.10.

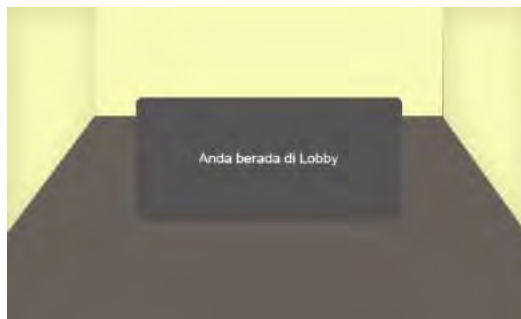


Gambar 3.9 Rancangan Antarmuka Pilihan Destinasi



Gambar 3.10 Rancangan Tampilan *Virtual Path*

Kemudian pengguna dapat mengikuti *virtual path* yang terbentuk untuk menuju ke destinasi. Pada aplikasi ini tiap ruangan akan memiliki penanda yang berfungsi menampilkan informasi ruangan. Ketika pengguna aplikasi memasuki ruangan, maka akan muncul informasi ruangan yang menandakan pengguna sedang berada pada ruangan tersebut. Rancangan tampilan informasi ruangan akan terlihat pada Gambar 3.11.



Gambar 3.11 Rancangan Tampilan Informasi Ruangan

3.2.8 Perancangan Skenario Tutorial

Skenario tutorial dibutuhkan sebagai pengenalan awal terhadap pengguna tentang fitur-fitur yang disediakan oleh aplikasi serta cara penggunaan aplikasi. Skenario tutorial dirancang berdasarkan pada kasus penggunaan dan fitur yang ada pada aplikasi. Pada skenario tutorial, pengguna akan diberi suatu *task* yang harus diselesaikan oleh pengguna. Terdapat 10 *task* yang ada pada skenario tutorial, dimana tiap *task* tersebut saling berurutan. Rincian *task* yang akan diberikan adalah sebagai berikut.

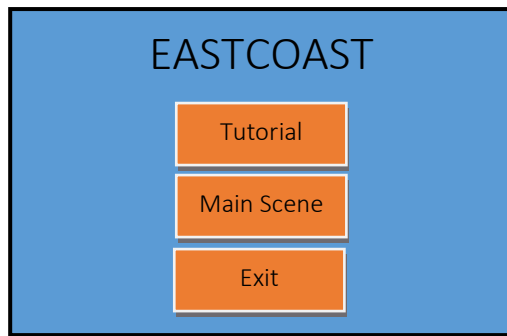
- Task 1 : Mencari Menu Navigasi
- Task 2 : Menggunakan Fitur *Navigation (Path)* Untuk Mencari Lokasi Dalam Lantai yang Sama
- Task 3 : Mengikuti *Path* yang Terbentuk Untuk Menuju Lokasi Dalam Lantai yang Sama
- Task 4 : Menggunakan Fitur *Navigation (Path)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator
- Task 5 : Mengikuti *Path* Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Eskalator
- Task 6 : Menggunakan Fitur *Navigation (Path)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift
- Task 7 : Mengikuti *Path* Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Lift
- Task 8 : Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Sama
- Task 9 : Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator
- Task 10: Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift

Setelah pengguna menyelesaikan semua *task* yang telah diberikan, pengguna akan diarahkan ke skenario utama secara otomatis. Dengan diberikannya gambaran tentang fitur-fitur yang

ada pada aplikasi melalui skenario tutorial, diharapkan pengguna tidak bingung untuk menggunakan fitur yang ada saat berada pada skenario utama.

3.2.9 Perancangan Skenario Menu Utama

Skenario Menu Utama merupakan skenario awal yang dijalankan pengguna saat menggunakan aplikasi. Pada skenario menu utama terdapat tiga tombol yang disediakan, yaitu tombol “Tutorial”, “Main Scene”, dan “Exit”. Tombol “Tutorial” digunakan untuk mengarahkan pengguna menuju skenario tutorial, Tombol “Main Scene” digunakan untuk mengarahkan pengguna menuju skenario utama, dan tombol “Exit” digunakan untuk keluar dari aplikasi. Rancangan antarmuka skenario menu utama dapat dilihat pada Gambar 3.12.



Gambar 3.12 Rancangan Antarmuka Skenario Menu Utama

(Halaman ini sengaja dikosongkan)

BAB IV IMPLEMENTASI

Pada bab ini dibahas mengenai implementasi yang dilakukan berdasarkan rancangan yang telah dijabarkan pada bab sebelumnya.

4.1 Lingkungan Implementasi

Lingkungan implementasi merupakan lingkungan dimana aplikasi akan dibangun. Lingkungan implementasi dibagi menjadi dua, yaitu lingkungan implementasi berupa perangkat keras, dan lingkungan implementasi berupa perangkat lunak.

4.1.1 Perangkat Keras

Perangkat keras yang digunakan dalam pengembangan aplikasi ini adalah komputer dengan spesifikasi sebagai berikut :

- Tipe : ASUS A450L
- Prosesor : Intel (R) Core(TM) i5-4200U @1.60GHz (4CPUs), ~2.3 GHz
- Memory : 8192 MB

Sedangkan perangkat keras yang digunakan untuk implementasi adalah perangkat bergerak dengan spesifikasi sebagai berikut :

- Tipe : Xiaomi Mi4i
- Prosesor : Qualcomm Snapdragon 615, Octa-Core 64-bit ARM Cortex A-53
- Memory : 2048 MB

4.1.2 Perangkat Lunak

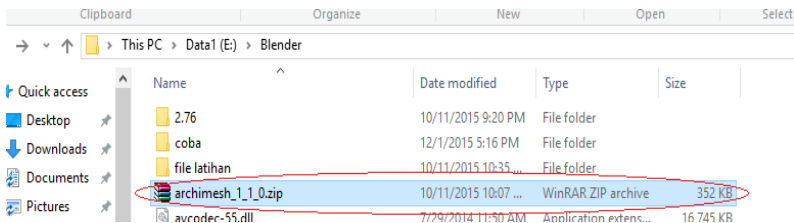
Perangkat lunak yang digunakan dalam pengembangan aplikasi ini adalah sebagai berikut :

- Sistem Operasi : Microsoft Windows 10 Pro 64-bit

- Perangkat Pengembang : Blender 2.7.6, Unity 5.3.4f1 (64-bit), Microsoft Visual Studio 2015
 - SDK : Cardboard SDK
- Sedangkan perangkat lunak yang digunakan untuk implementasi memiliki spesifikasi sebagai berikut :
- Sistem Operasi : Android OS, v5.0.2 (Lollipop)

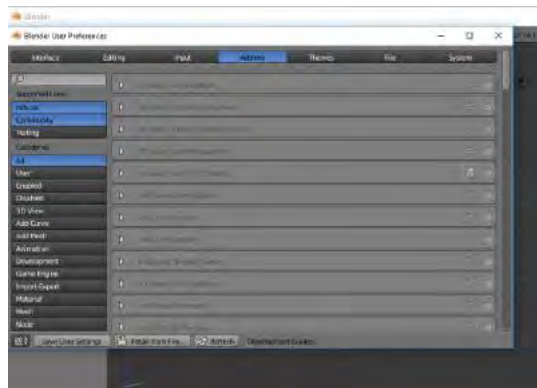
4.2 Implementasi Lingkungan Virtual Mall

Pada tahap ini, hal yang pertama kali dilakukan adalah membuat objek ruangan dengan menggunakan *software* Blender. Untuk memudahkan pembuatan ruangan pada *software* Blender, penulis menggunakan sebuah *add-on* yaitu Archimesh. Archimesh adalah sebuah *add-on* yang disediakan *software* Blender untuk membuat elemen-elemen arsitektur seperti ruangan, rumah, tangga, pintu, jendela, dan sebagainya. *Add-on* Archimesh pada Blender dapat diunduh pada website <https://github.com/Antonioya/blender/tree/master/archimesh>. File *add-on* Archimesh setelah diunduh akan tampak pada Gambar 4.1.



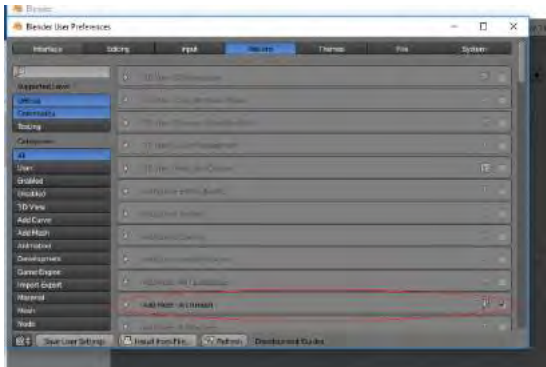
Gambar 4.1 File Add-on Archimesh

Setelah berhasil mengunduh file *add-on* Archimesh, langkah selanjutnya adalah membuka *software* Blender untuk memulai instalasi *add-on* pada Blender. Caranya adalah dengan mengklik menu 'File' lalu memilih 'User Preferences'. Kemudian dilanjutkan dengan memilih tab 'Add-ons' dan akan muncul tampilan seperti pada Gambar 4.2.



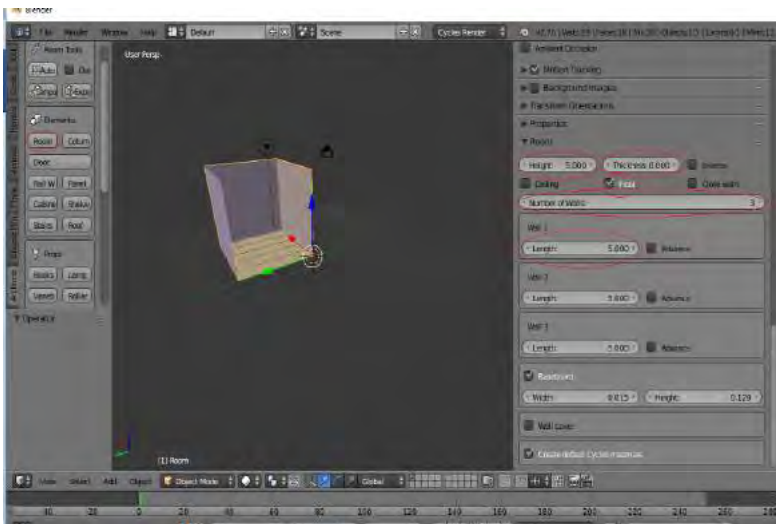
Gambar 4.2 Tampilan ‘User Preferences’ sebelum terinstall add-on Archimesh

Langkah berikutnya adalah mengklik tombol ‘*Install from File..*’, lalu cari file *add-on* Archimesh yang sebelumnya telah diunduh. Setelah file *add-on* Archimesh telah dipilih, dilanjutkan dengan mengklik tombol ‘*Install from File..*’ dan *add-on* Archimesh akan ter-*install*. Untuk mengetahui apakah *add-on* Archimesh telah ter-*install* atau belum dapat dilihat pada menu ‘*User Preferences*’ seperti yang terlihat pada Gambar 4.3. Apabila *add-on* Archimesh telah ter-*install*, maka akan muncul centang pada sebelah kanan tulisan Archimesh.



Gambar 4.3 Tampilan ‘User Preferences’ setelah ter-install add-on Archimesh

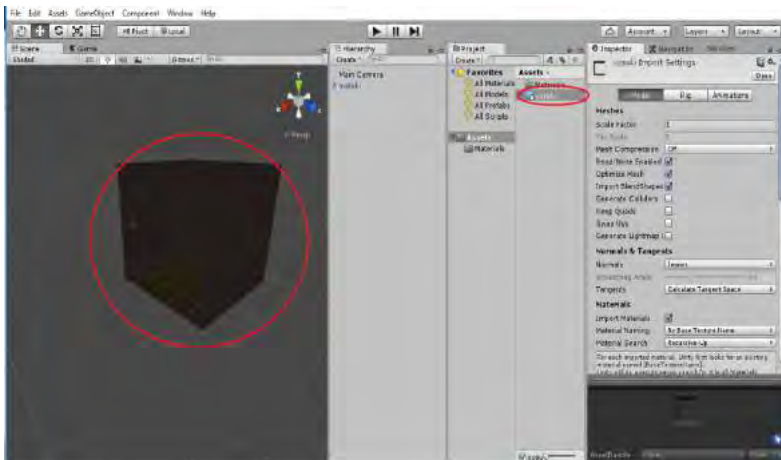
Setelah *add-on* Archimesh ter-*install*, langkah berikutnya adalah memulai membuat objek ruangan pada Blender. Ruangan pertama yang akan dibuat adalah ruangan “XOSUKI”. Langkah pertama adalah mengklik tab ‘Archimesh’ pada menu sebelah kiri, kemudian mengklik tombol ‘Room’ dan satu dinding ruangan akan terbentuk. Kemudian mengatur tinggi dan ketebalan ruangan pada kolom ‘Height’ dan ‘Thickness’. Kemudian dilanjutkan dengan menambahkan lantai pada ruangan dengan memberi centang pada ‘Floor’. Lalu mengatur jumlah dinding yang dibutuhkan pada ruangan pada kolom ‘Number of Walls’ dan mengatur panjang dinding yang dibentuk pada kolom ‘Length’ di setiap dinding. Tampilan pembuatan ruangan akan terlihat pada Gambar 4.4.



Gambar 4.4 Pembuatan ruangan pada Blender

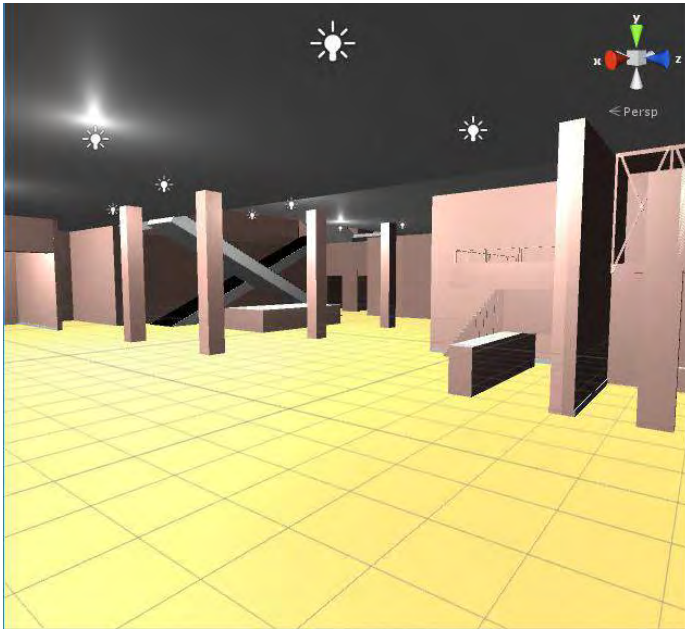
Setelah ruangan terbentuk, kemudian menyimpan file blender ruangan yang telah dibuat dengan ekstensi “.blend” agar dapat di-*import* ke Unity. Caranya dengan mengklik menu ‘File’, kemudian ‘Save As’, dan dilanjutkan dengan memberi nama file dan mengklik tombol ‘Save As Blender File’. Setelah file ruangan

dimiliki, langkah berikutnya adalah memasukkan file Blender yang telah dibuat ke dalam Unity. Caranya dengan drag file “.blend” yang telah dibuat pada folder “Assets” di *project* Unity yang telah terbuka. Kemudian drag *asset* yang telah ter-import pada *Scene* untuk membuat lingkungan virtual pada Unity. Tampilan ruangan setelah dimasukkan ke dalam Unity akan terlihat pada Gambar 4.5.



Gambar 4.5 Tampilan ruangan setelah dimasukkan ke dalam Unity

Kemudian dilanjutkan dengan pembuatan ruangan yang lain pada mall dengan langkah yang sama. Pengaturan letak ruangan dilakukan pada Unity, berdasarkan pada denah yang ada. Setelah semua ruangan terbentuk dan posisi ruangan pada lingkungan virtual telah diatur, dilanjutkan dengan memberi pencahayaan pada mall dengan mengklik menu ‘*GameObject*’>‘*Light*’. Pada aplikasi ini jenis pencahayaan yang digunakan adalah satu *Directional Light*, dengan penambahan *Point Light* pada area yang dirasa kurang pencahayaan. Tampilan lingkungan virtual mall setelah diberi pencahayaan akan terlihat pada Gambar 4.6.

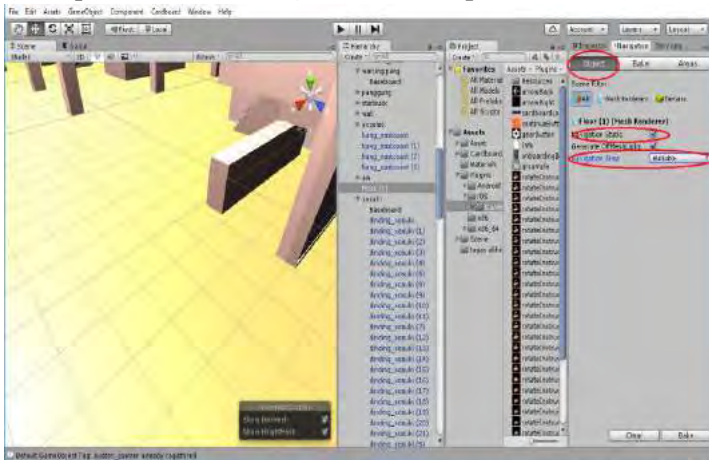


Gambar 4.6 Tampilan Lingkungan Virtual Mall Setelah Diberi Pencahayaan

4.3 Implementasi Area Navigasi

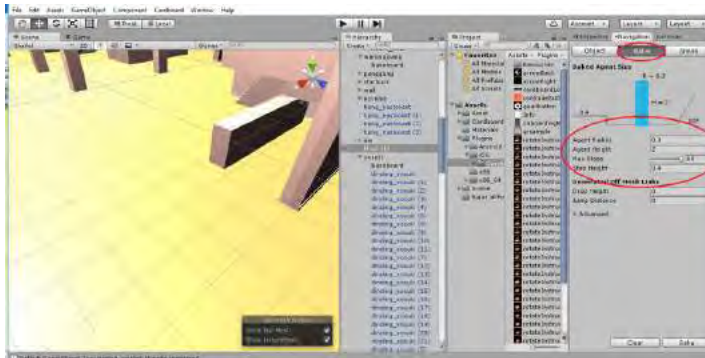
Pada tahap ini akan dijelaskan implementasi area navigasi yang dibuat pada aplikasi. Untuk memulai membuat area navigasi, *window* 'Navigation' harus diaktifkan terlebih dahulu dengan cara mengklik pada 'Window' > 'Navigation'. Setelah *window* Navigation muncul, maka dapat diatur area mana yang dapat dilalui pengguna, serta area mana yang tidak dapat dilalui oleh pengguna pada tab 'Object'. Langkah pertama yang harus dilakukan adalah memilih elemen ruangan yang termasuk dari area navigasi. Kemudian pada *window* 'Navigation' tab 'Object', lakukan centang pada Navigation Static, dan mengatur apakah elemen ruangan tersebut dapat dilalui oleh pengguna atau tidak pada bagian *Navigation Area*. Apabila pengguna dapat melewati

elemen ruangan tersebut, semisal pada objek lantai, maka yang dipilih pada bagian *Navigation Area* adalah *Walkable*. Sebaliknya, apabila pengguna tidak dapat melewati elemen ruangan tersebut, semisal pada objek dinding, maka yang dipilih pada bagian *Navigation Area* adalah *Not Walkable*. Tampilan tab ‘Object’ pada window *Navigation* terlihat pada Gambar 4.7.



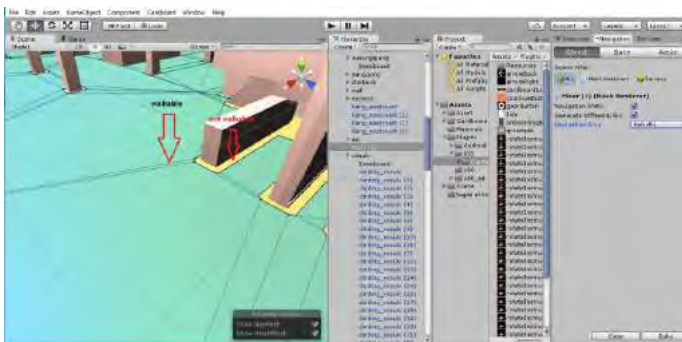
Gambar 4.7 Tampilan Tab ‘Object’ pada Window Navigation

Langkah berikutnya adalah mengatur *bake settings* pada tab ‘Bake’ untuk menyesuaikan area navigasi yang akan dibuat dengan objek pengguna. *Agent Radius* digunakan untuk mengatur seberapa dekat objek pengguna dapat menyentuh dinding atau objek yang tidak dapat dilalui pengguna. Semakin besar nilai dari *Agent Radius*, maka semakin jauh pula jarak antara pengguna dengan objek yang tidak dapat dilalui pengguna. *Agent Height* digunakan untuk mengatur tinggi maksimal dari objek yang dapat dilalui objek pengguna. *Max Slope* digunakan untuk mengatur seberapa besar sudut kemiringan yang dapat dilalui objek pengguna. Dan *Step Height* digunakan untuk mengatur seberapa tinggi objek yang dapat dilalui pengguna, biasanya digunakan pada tangga. Tampilan tab ‘Bake’ pada window *Navigation* terlihat pada Gambar 4.8.



Gambar 4.8 Tampilan Tab ‘Bake’ pada Window Navigation

Kemudian lakukan langkah-langkah diatas pada setiap elemen ruangan pada mall untuk mengatur area navigasi pada mall secara keseluruhan. Setelah semua elemen ruangan selesai diatur, selanjutnya adalah mengklik tombol ‘Bake’ untuk membangun *Navigation Mesh* yang merupakan area navigasi dari aplikasi. Dapat dilihat pada Scene bahwa daerah yang berwarna biru menandakan bahwa objek pengguna dapat melewati area tersebut, sedangkan yang tidak diberi warna biru menandakan bahwa objek pengguna tidak dapat melewati area tersebut. Tampilan *Navigation Mesh* yang telah terbentuk terlihat pada Gambar 4.9.



Gambar 4.9 Tampilan Navigation Mesh yang telah terbentuk

4.4 Implementasi Objek Pengguna

Pada tahap ini akan dijelaskan mengenai implementasi dari objek pengguna. Langkah pertama adalah pembuatan objek pengguna berupa *capsule* dengan mengklik pada menu ‘*GameObject*’ > ‘*3D Object*’ > ‘*Capsule*’. Tampilan dari objek pengguna yang telah dibuat dapat dilihat pada Gambar 4.10.



Gambar 4.10 Tampilan Objek Pengguna

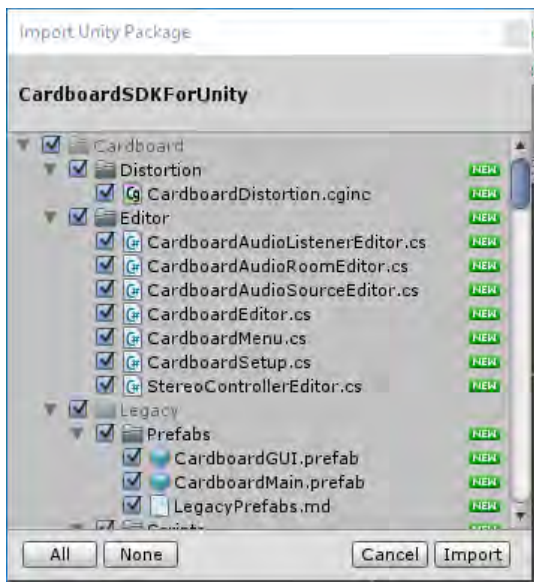
Implementasi dari objek pengguna meliputi integrasi Google Cardboard dengan Unity, agar pengguna dapat melihat lingkungan sekitar dengan bantuan perangkat Google Cardboard. Kemudian dilanjutkan dengan implementasi kontrol pengguna, yaitu pemberian *script* pada objek pengguna agar pengguna dapat berjalan dalam lingkungan virtual. Dan yang terakhir adalah implementasi NavMesh Agent, yaitu pendefinisian objek pengguna sebagai NavMesh Agent agar pengguna hanya dapat berjalan pada area navigasi yang telah terbentuk.

4.4.1 Implementasi Integrasi Google Cardboard dengan Unity

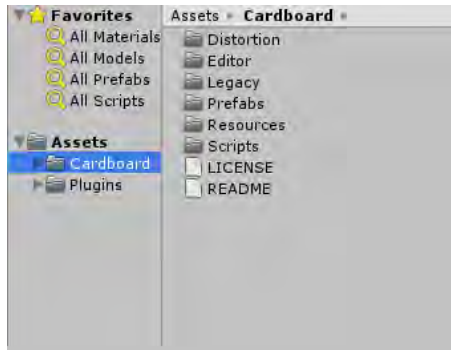
Implementasi integrasi perangkat Google Cardboard dengan Unity memerlukan suatu penghubung, dalam hal ini adalah Cardboard SDK. Cardboard SDK dapat diunduh pada situs <https://developers.google.com/vr/unity/>. Hasil unduh dari SDK ini

berupa package untuk mengintegrasikan perangkat Google Cardboard dengan Unity. Package tersebut kemudian diimpor kedalam Unity. Proses impor dapat dilihat pada Gambar 4.11. Setelah proses impor berhasil, maka akan muncul dua folder baru di bagian *Assets* pada Unity. Hasil proses tersebut dapat dilihat pada Gambar 4.12.

Dua folder tersebut adalah folder *Cardboard* dan *Plugins*. Folder *Cardboard* berisikan asset-asset yang digunakan untuk membuat aplikasi dengan memanfaatkan Google Cardboard termasuk kamera yang nantinya akan dipasangkan pada objek pengguna dan menggantikan *Main Camera* yang ada pada Unity, agar pengguna dapat melihat lingkungan sekitar dalam lingkungan virtual dengan menggunakan perangkat Google Cardboard. sedangkan folder *Plugins* berisi pengaturan pada Google Cardboard.



Gambar 4.11 Proses Impor *Package* Integrasi Google Cardboard ke Unity



Gambar 4.12 Hasil Proses Impor Package Integrasi Google Cardboard

Langkah berikutnya adalah memasang kamera Google Cardboard pada objek pengguna, menggantikan *Main Camera* yang ada pada Unity. Langkah pertama adalah menghapus *Main Camera* yang ada pada 'Hierarchy' terlebih dahulu. Kemudian pasang *Cardboard Main* yang terdapat pada folder '*Cardboard*' > '*Prefabs*' ke dalam Capsule. Tidak lupa untuk menambahkan *Cardboard Reticle* pada *Main Camera* yang baru sebagai pointer pada tampilan realitas virtual. Cardboard Reticle dapat diambil dari folder '*Cardboard*' > '*Prefabs*' > '*UI*'. Hasil proses dari penambahan kamera Cardboard akan terlihat pada Gambar 4.13.



Gambar 4.13 Hasil Proses Penambahan Kamera Cardboard

4.4.2 Implementasi Kontrol Pengguna

Kontrol pengguna dalam aplikasi ini memanfaatkan *trigger* yang ada pada Google Cardboard. Dalam aplikasi ini, objek pengguna akan berjalan secara *automatic* ketika *trigger* ditekan. Dan ketika *trigger* ditekan sekali lagi, maka objek pengguna akan berhenti. Diperlukan sebuah *script* untuk mengatur pergerakan objek pengguna, yaitu “Autowalk.cs”. *Script* tersebut akan diletakkan pada objek pengguna, agar objek pengguna dapat bergerak ketika *trigger* diaktifkan. Fungsi untuk pergerakan pengguna dapat dilihat pada Kode Sumber 4.1.

```
using UnityEngine;
using System.Collections;
public class Autowalk : MonoBehaviour {
private const int RIGHT_ANGLE = 90;
    // This variable determinates if the player will move or not
    public bool isWalking = false;
    CardboardHead head = null;
    //This is the variable for the player speed
    public float speed = 1;
    public bool walkWhenTriggered = true;
    public bool walkWhenLookDown;
    public double thresholdAngle;
    public bool freezeYPosition;
    public float yOffset;
    void Start()
    {
        head = Camera.main.GetComponent<StereoController>().Head;
    }
    public void SetWalkingFalse()
    {
        isWalking = false;
    }
    void Update()
    {
        print(head.transform.eulerAngles.y);
        // Walk when the Cardboard Trigger is used
        if (walkWhenTriggered && !walkWhenLookDown && !isWalking
&& Cardboard.SDK.Triggered)
        {
            isWalking = true;
        }
    }
}
```

```

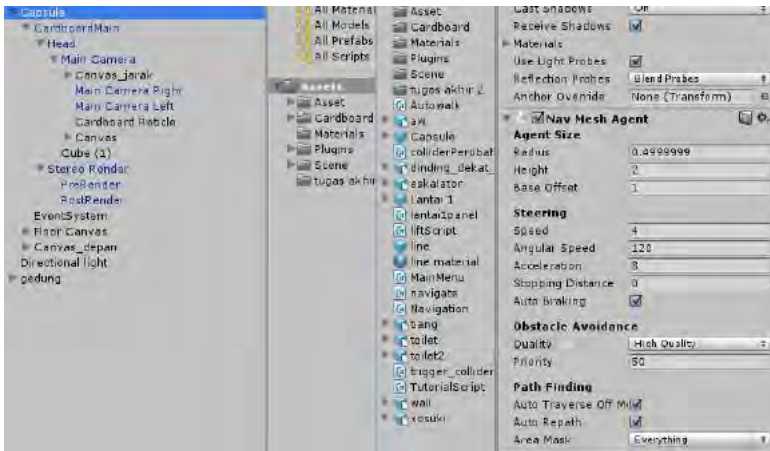
        else if (walkWhenTriggered && !walkWhenLookDown &&
isWalking && Cardboard.SDK.Triggered)
        {
            isWalking = false;
        }
        if (isWalking)
        {
            Vector3 direction = new
Vector3(head.transform.forward.x,
head.transform.forward.z).normalized * speed * Time.deltaTime;
            Quaternion rotation = Quaternion.Euler(new Vector3(0,
-head.transform.rotation.eulerAngles.y, 0));
            transform.Translate(rotation * direction);
        }
        if (freezeYPosition)
        {
            transform.position = new Vector3(transform.position.x,
yOffset, transform.position.z);
        }
    }
}

```

Kode Sumber 4.1 Implementasi Kontrol Pergerakan Pengguna

4.4.3 Implementasi Objek Sebagai *NavMesh Agent*

Pada tahap ini akan dijelaskan tentang pendefinisian objek pengguna sebagai NavMesh Agent. Hal ini dibutuhkan agar objek pengguna hanya dapat bergerak pada *Navigation Mesh* yang telah terbentuk, begitu juga dengan navigasi. Navigasi hanya dapat dilakukan pada lokasi yang berada pada *Navigation Mesh*. Pendefinisian objek pengguna sebagai *NavMesh Agent* dilakukan dengan cara mengklik objek pengguna pada *Scene* atau *Hierarchy* yaitu *Capsule*, kemudian mengklik tombol ‘Add Component’ pada menu *Inspector*, dan mengetik *NavMesh Agent* pada field yang disediakan, kemudian tekan *Enter*. Hasil dari proses pendefinisian objek pengguna sebagai NavMesh Agent akan terlihat pada Gambar 4.14.



**Gambar 4.14 Hasil Proses Pendefinisian Objek Pengguna
Sebagai NavMesh Agent**

4.5 Implementasi Informasi Ruangan

Pada aplikasi ini, tiap ruangan memiliki *collider*, dimana apabila objek pengguna memasuki *collider* ruangan, maka aplikasi akan mengaktifkan *trigger* untuk menampilkan *panel* yang menandakan informasi yang berupa nama dari ruangan yang dimasuki. Ketika pengguna sudah keluar dari area *collider* ruangan, maka sistem akan mengaktifkan *trigger* untuk menonaktifkan informasi ruangan. Potongan kode sumber untuk implementasi informasi ruangan dapat dilihat pada Kode Sumber 4.2 yang diberi nama *trigger_collider.cs* dan tampilan informasi ruangan dapat dilihat pada Gambar 4.15.

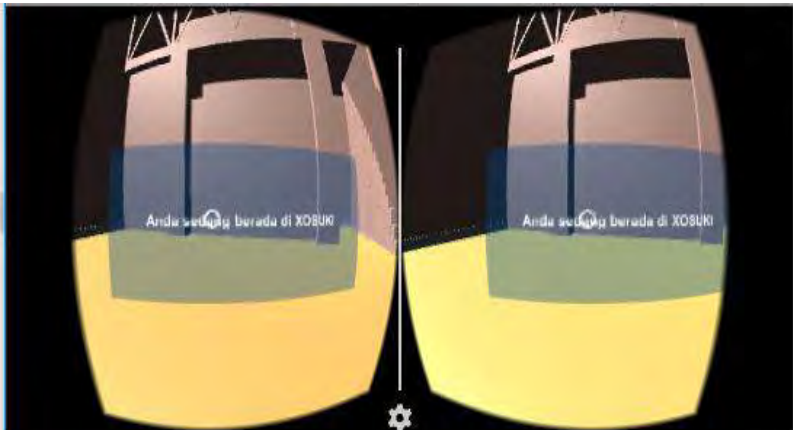
```
using UnityEngine;
using System.Collections;
using UnityEngine.UI;
public class trigger_collider : MonoBehaviour {
    public Text text1;
    public GameObject notif;
    IEnumerator OnTriggerEnter(Collider other)
    {
```

```

    if (other.gameObject.name == "xosuki_collider" ||
other.gameObject.name == "xosuki_pointer")
    {
        notif.gameObject.SetActive(true);
        text1.text = "Anda sedang berada di XOSUKI";
        yield return new WaitForSeconds(5);
        notif.gameObject.SetActive(false);
        text1.text = "";
    }
}
void OnTriggerExit (Collider other)
{
    notif.gameObject.SetActive(false);
    text1.text = "";
}
}
}

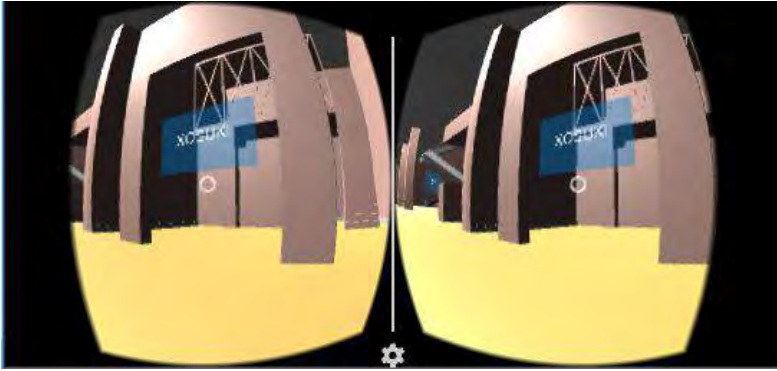
```

Kode Sumber 4.2 Implementasi Informasi Ruangan



Gambar 4.15 Tampilan Informasi Ruangan

Selain *collider* pada tiap ruangan, terdapat *panel* yang selalu aktif pada tiap ruangan sebagai penanda ruangan agar pengguna dapat melihat nama ruangan yang ada dalam lingkungan virtual dari jarak pandang pengguna. Tampilan panel penanda ruangan dapat dilihat pada Gambar 4.16.



Gambar 4.16 Tampilan Panel Penanda Ruangan

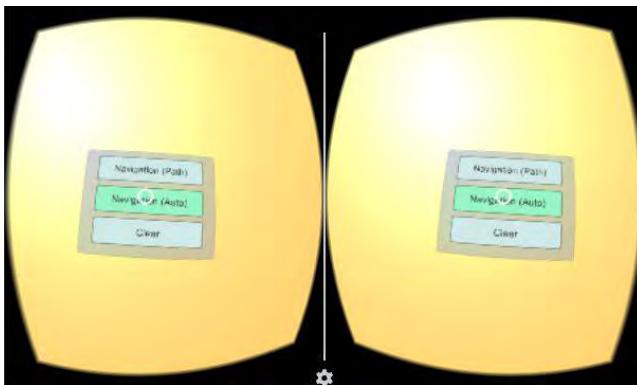
4.6 Implementasi Sistem Navigasi

Pada tahap ini akan dijelaskan implementasi dari sistem navigasi yang dibuat. Implementasi sistem navigasi meliputi implementasi menu navigasi, implementasi fitur *Navigation(path)*, implementasi fungsi eskalator, implementasi fungsi lift, dan implementasi fitur *Navigation(Auto)*.

4.6.1 Implementasi Menu Navigasi

Langkah pertama yang dilakukan adalah membuat menu untuk memilih jenis navigasi yang berada dibawah pengguna. Terdapat tiga buah tombol yang terdapat pada menu navigasi, yaitu tombol “Navigation (path)”, tombol “Navigation(auto)”, dan tombol “Clear”. *Class* yang digunakan pada implementasi ini adalah *class* navigate. Pada *class* navigate terdapat variabel *navigatePath* untuk menentukan jenis navigasi. Apabila nilai *navigatePath* adalah 1, maka jenis navigasi yang dipilih adalah *Navigation(Path)*. Apabila nilai *navigatePath* adalah 2, maka jenis navigasi yang dipilih adalah *Navigation (Auto)*. Ketika tombol “Navigation (path)” dipilih, maka sistem akan menjalankan fungsi *SetNavigationPath* untuk mengubah nilai variabel *navigatePath* menjadi 1 dan kemudian mengarahkan pengguna menuju menu pemilihan rute. Ketika tombol

“Navigation (auto)” dipilih, maka sistem akan menjalankan fungsi *SetNavigationPathFalse* untuk mengubah nilai variabel *navigatePath* menjadi 2 dan kemudian mengarahkan pengguna menuju menu pemilihan rute. Dan ketika tombol “Clear” dipilih, maka sistem akan menjalankan fungsi *ResetPath* dan rute yang telah terbentuk pada aplikasi akan di-reset. Pada tiga tombol tersebut juga terdapat fungsi *SetDestinationFalse* untuk me-reset ulang nilai destinasi. Tampilan menu navigasi akan terlihat pada Gambar 4.17. Potongan kode sumber untuk fungsi *SetNavigationPath* dapat dilihat pada Kode Sumber 4.3 dan potongan kode sumber untuk fungsi *SetNavigationPathFalse* dapat dilihat pada Kode Sumber 4.4. Potongan kode sumber untuk fungsi *ResetPath* dapat dilihat pada Kode Sumber 4.5 dan potongan kode sumber untuk fungsi *SetDestinationFalse* dapat dilihat pada Kode Sumber 4.6.



Gambar 4.17 Tampilan Menu Navigasi

```
public void SetNavigationPath()
{
    navigatePath = 1;
    jarakObject.transform.localScale = new Vector3(0, 0, 0);
    jarakValue.transform.localScale = new Vector3(0, 0, 0);
    SetDestinationFalse();
}
```

Kode Sumber 4.3 Fungsi *SetNavigationPath*

```

public void SetNavigationPathFalse()
{
    navigatePath = 2;
    jarakObject.transform.localScale = new Vector3(0, 0, 0);
    jarakValue.transform.localScale = new Vector3(0, 0, 0);
    SetDestinationFalse();
}

```

Kode Sumber 4.4 Fungsi *SetNavigationPathFalse*

```

public void ResetPath()
{
    agent.ResetPath();
    Destroy(line);
    Destroy(line2);
    navigatePath = 0;
    jarakObject.transform.localScale = new Vector3(0, 0, 0);
    jarakValue.transform.localScale = new Vector3(0, 0, 0);
    SetFalse();
}
public void SetFalse()
{
    navigatePath = 0;
    liftCase = false;
    SetDestinationFalse();
}

```

Kode Sumber 4.5 Fungsi *ResetPath*

```

public void SetDestinationFalse()
{
    xosuki_direction = false;
    aw_direction = false;
    solaria_direction = false;
    noodlehouse_direction = false;
    informa_direction = false;
    starbuck_direction = false;
    excelso_direction = false;
    houseofwok_direction = false;
    warungipang_direction = false;
    mokko_direction = false;
    crystaljade_direction = false;
    hypermart_direction = false;
    boston_direction = false;
    jmtop_direction = false;
    century_direction = false;
    head2toe_direction = false;
    informa2_direction = false;
}

```



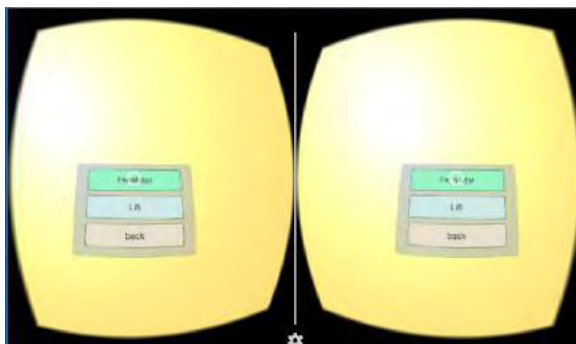
```

    funworld_direction = false;
    shinjuku_direction = false;
    ellery_direction = false;
    yopiesalon_direction = false;
}

```

Kode Sumber 4.6 Fungsi *SetDestinationFalse*

Setelah berada di menu pemilihan rute, pengguna diharuskan memilih rute yang akan dilewati untuk menuju ke destinasi. Terdapat tiga buah tombol yang terdapat pada menu pemilihan rute, yaitu tombol “Eskalator”, tombol ”Lift”, dan tombol “back”. Ketika tombol “Eskalator” dipilih, maka sistem akan menjalankan fungsi *SetLiftCaseFalse* dan kemudian mengarahkan pengguna menuju menu pemilihan destinasi. Ketika tombol “Lift” dipilih, maka sistem akan menjalankan fungsi *SetLiftCase* yang terdapat pada dan kemudian mengarahkan pengguna menuju menu pemilihan destinasi. Ketika tombol “back” dipilih, maka sistem akan mengarahkan pengguna ke menu pemilihan jenis navigasi. Menu pemilihan destinasi akan muncul di depan pengguna. Tampilan menu pemilihan rute akan terlihat pada Gambar 4.18. Potongan kode sumber untuk fungsi *SetLiftCaseFalse* dapat dilihat pada Kode Sumber 4.7 dan potongan kode sumber untuk fungsi *SetLiftCase* dapat dilihat pada Kode Sumber 4.8.



Gambar 4.18 Tampilan Menu Pemilihan Rute

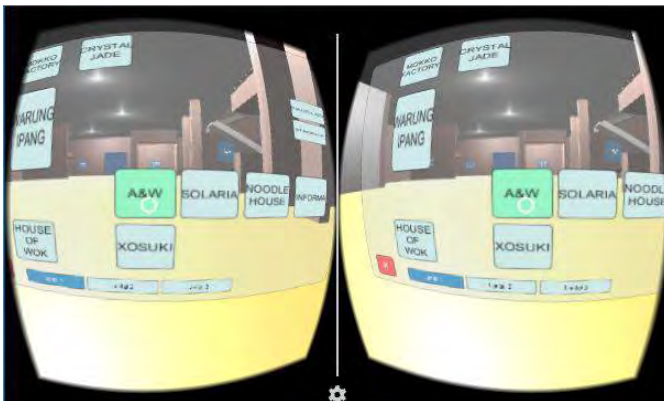
```
public void SetLiftCaseFalse()
{
    liftCase = false;
}
```

Kode Sumber 4.7 Fungsi *SetLiftCaseFalse*

```
public void SetLiftCase()
{
    liftCase = true;
}
```

Kode Sumber 4.8 Fungsi *SetLiftCase*

Setelah berada pada menu pemilihan destinasi, pengguna diharuskan untuk memilih destinasi yang dituju dengan mengklik salah satu tombol yang terdapat pada menu. Ketika salah satu tombol destinasi diklik, maka sistem akan menjalankan fungsi untuk mengatur destinasi sesuai destinasi yang dipilih. Semisal destinasi yang dipilih adalah “XOSUKI”, maka fungsi yang dijalankan adalah *SetXosukiDirection*. Tampilan menu pemilihan destinasi akan terlihat pada Gambar 4.19. Potongan kode sumber untuk fungsi mengatur destinasi akan terlihat pada Kode Sumber 4.9.



Gambar 4.19 Tampilan Menu Pemilihan Destinasi

```

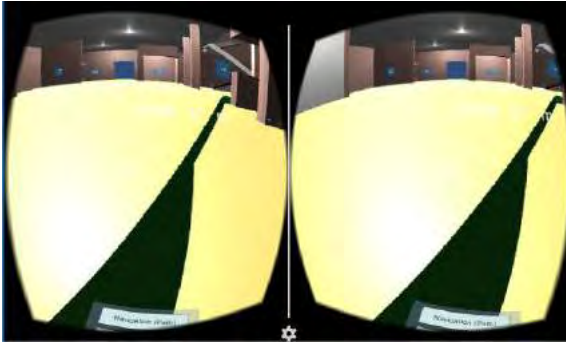
public void SetXosukiDirection()
{
    agent.ResetPath();
    Destroy(line);
    Destroy(line2);
    tujuan = GameObject.FindGameObjectWithTag("xosuki_pointer");
    print(tujuan.gameObject.tag);
    destinationFloor = 0;
    SetDestinationFalse();
    xosuki_direction = true;
    if(navigatePath == 2)
    {
        AutoPath();
    }
}

```

Kode Sumber 4.9 Fungsi Untuk Mengatur Pilihan Destinasi

4.6.2 Implementasi Fitur Navigation(Path)

Fitur *Navigation(Path)* adalah fitur untuk melakukan navigasi dengan menampilkan garis antara posisi pengguna dan posisi destinasi yang menggambarkan rute terdekat yang harus dilalui pengguna untuk menuju destinasi. Garis akan ditampilkan setelah pengguna memilih destinasi pada menu pemilihan destinasi. Selanjutnya sistem akan melakukan navigasi sesuai dengan pilihan yang telah dipilih. Apabila saat pemilihan jenis navigasi pengguna memilih fitur “Navigation(path)”, maka sistem akan melakukan navigasi dengan menggambar garis antara posisi pengguna dengan posisi destinasi. Selain garis, jarak untuk menuju ke destinasi juga akan ditampilkan. Semakin kecil jarak maka semakin dekat posisi pengguna menuju destinasi. Penggambaran garis memanfaatkan fitur *Line Renderer* pada Unity. Setiap tiga detik sekali, garis dan jarak untuk menuju ke destinasi akan ter-*update*. Fungsi untuk menggambar garis terdapat adalah *FindPath*, dan pemanggilan fungsi tersebut berada pada fungsi *FixedUpdate*. Tampilan garis navigasi yang terbentuk dapat dilihat pada Gambar 4.20. Potongan kode sumber untuk implementasi fitur “Navigation(path)” dapat dilihat pada Kode Sumber 4.10.



Gambar 4.20 Tampilan Garis Navigasi

```
IEnumerator Jeda()
{
    yield return new WaitForSeconds(1);
}
void FindPath()
{
    if(lineLama != null)
    {
        Destroy(line);
        Destroy(line2);
    }
    agent.CalculatePath(tujuan.transform.position, path);
    lastGameObject = GameObject.Instantiate(penanda,
    transform.position, Quaternion.identity) as GameObject;
    lineLama = lastGameObject;
    line = lastGameObject.GetComponent<LineRenderer>();
    line.SetVertexCount(path.corners.Length);
    jarak =
    Vector3.Distance(lastGameObject.transform.position,
    path.corners[0]);
    total = jarak;
    for (int i = 0; i < path.corners.Length; i++)
    {
        if (i + 1 <= path.corners.Length - 1)
        {
            jarak = Vector3.Distance(path.corners[i], path.corners[i +
1]);
            total += jarak;
        }
        line.SetPosition(i, path.corners[i]);
    }
}
```

```

total = Mathf.Ceil(total);
jarakObject.transform.localScale = new Vector3(1, 1, 1);
jarakValue.text = total.ToString();
jarakValue.transform.localScale = new Vector3(1, 1, 1);
}
void FixedUpdate ()
{
    if (xosuki_direction == true && elapsed <= 0)
    {
        if(navigatePath == 1)
        {
            StartCoroutine("Jeda");
            FindPath();
            elapsed = 3.0f;
        }
    }
    elapsed -= Time.deltaTime;
}
}
}

```

Kode Sumber 4.10 Implementasi Fitur *Navigation(Path)*

Dalam melakukan penggambaran garis navigasi, langkah pertama adalah melakukan pencarian rute terdekat dan melakukan pengecekan apakah rute yang ditemukan dapat dilalui atau tidak dengan menggunakan fungsi *CalculatePath*. Kemudian dilanjutkan dengan memanggil objek pada posisi pengguna berdiri sekarang sebagai titik garis pertama. Kemudian dilakukan perulangan, pada setiap *path corner* yang ditemui dalam pencarian rute sebelumnya akan dijadikan titik dari *line renderer* yang dibuat. Jarak tempuh juga bisa didapatkan melalui akumulasi dari setiap perulangan tersebut.

4.6.3 Implementasi Fungsi Eskalator

Kemudian untuk menyelesaikan permasalahan destinasi yang berbeda lantai dengan rute melewati eskalator, yang pertama kali dilakukan adalah melakukan pengecekan posisi pengguna sekarang berada di lantai berapa. Pengecekan posisi pengguna menggunakan fungsi *CheckPosition* berdasarkan koordinat y objek pengguna. Kemudian pada eskalator juga diberi *collider* untuk menandakan apakah pengguna sedang berada di eskalator

naik atau eskalator turun. Setelah diketahui posisi objek pengguna, langkah selanjutnya adalah memberi fungsi pada eskalator, seperti ketika objek pengguna sedang berada dilantai satu, maka eskalator naik akan terbuka, dan eskalator turun akan tertutup. Agar navigasi tidak salah rute, digunakanlah *NavMeshObstacle* untuk membuka tutup eskalator. Potongan kode untuk fungsi *CheckPosition* dapat dilihat pada Kode Sumber 4.11. Potongan kode untuk fungsi pada collider eskalator dapat dilihat pada Kode Sumber 4.12. Potongan kode untuk implementasi fungsi eskalator dapat dilihat pada Kode Sumber 4.13.

```
public void CheckPosition()
{
    if(agent.transform.position.y < -1.6 &&
agent.transform.position.y > -1.8)
    {
        playerFloor = 0;
        inEskalatorUp = false;
        inEskalatorDown = false;
        print("lantai 1");
    }
    else if (agent.transform.position.y < 8.9 &&
agent.transform.position.y > 8.8)
    {
        playerFloor = 1;
        inEskalatorUp = false;
        inEskalatorDown = false;
        print("lantai 2");
    }
    else if (agent.transform.position.y < 20.6 &&
agent.transform.position.y > 20.5)
    {
        playerFloor = 2;
        inEskalatorUp = false;
        inEskalatorDown = false;
        print("lantai 3");
    }
}
```

Kode Sumber 4.11 Fungsi *CheckPosition* Untuk Mengecek Posisi Pengguna

```

void OnTriggerEnter(Collider other)
{
    if (other.gameObject.tag == "collider_esk1_up")
    {
        inEskalatorUp = true;
    }
    else if (other.gameObject.tag == "collider_esk1_down")
    {
        inEskalatorDown = true;
    }
    else if (other.gameObject.tag == "collider_esk2_up")
    {
        inEskalatorUp = true;
    }
    else if (other.gameObject.tag == "collider_esk2_down")
    {
        inEskalatorDown = true;
    }
}

```

Kode Sumber 4.12 Fungsi *onTriggerEnter* pada *Collider* Eskalator

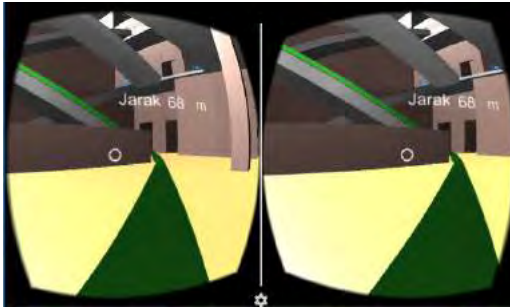
```

void FixedUpdate ()
{
    CheckPosition();
    if(playerFloor == 0 && inEskalatorUp == false &&
    inEskalatorDown == false){
        elevatorLt1Naik_up.GetComponent<NavMeshObstacle>().enabled =
        false;
        elevatorLt1Naik_bottom.GetComponent<NavMeshObstacle>().enabled =
        false;
        elevatorLt1Turun_up.GetComponent<NavMeshObstacle>().enabled =
        true;
        elevatorLt1Turun_bottom.GetComponent<NavMeshObstacle>().enabled =
        true;
        elevatorLt2Naik_up.GetComponent<NavMeshObstacle>().enabled =
        false;
        elevatorLt2Naik_bottom.GetComponent<NavMeshObstacle>().enabled =
        false;
        elevatorLt2Turun_up.GetComponent<NavMeshObstacle>().enabled =
        true;
        elevatorLt2Turun_bottom.GetComponent<NavMeshObstacle>().enabled =
        true;
    }
}

```

Kode Sumber 4.13 Implementasi Fungsi Eskalator

Untuk metode penggambaran garisnya, sama dengan implementasi fitur *Navigation(path)* yang dapat dilihat pada Kode Sumber 4.10. Tampilan garis navigasi melewati eskalator dapat dilihat pada Gambar 4.21.



Gambar 4.21 Tampilan Garis Navigasi Melewati Eskalator

4.6.4 Implementasi Fungsi Lift

Untuk menyelesaikan permasalahan destinasi yang berbeda lantai dengan rute melewati lift, langkah pertama yang dilakukan adalah menggambar garis dari posisi pengguna pada lift terdekat. Kemudian dilanjutkan dengan menggambar garis dari lift destinasi menuju destinasi. Pada tiap lift diberi penanda berupa objek, dan penanda tersebut dapat berfungsi juga sebagai *NavMesh Agent* karena yang dapat melakukan pencarian rute terdekat adalah *NavMesh Agent*. Sama seperti implementasi fungsi eskalator, diperlukan pengecekan posisi pengguna sekarang dan posisi destinasi. Potongan kode sumber untuk implementasi fitur *Navigation(path)* pada lantai yang berbeda melalui lift dapat dilihat pada Kode Sumber 4.14.

```
void FindPath()
{
    if(lineLama != null)
    {
        Destroy(line);
        Destroy(line2);
    }
    if(liftCase == true){
        if (playerFloor == destinationFloor){
```



```

        agent.CalculatePath(tujuan.transform.position,
path);
        lastGameObject = GameObject.Instantiate(penanda,
transform.position, Quaternion.identity) as GameObject;
        lineLama = lastGameObject;
        line =
        lastGameObject.GetComponent<LineRenderer>();
        line.SetVertexCount(path.corners.Length);
        jarak =
        Vector3.Distance(lastGameObject.transform.position,
path.corners[0]);
        total = jarak;
        for (int i = 0; i < path.corners.Length; i++){
            if (i + 1 <= path.corners.Length - 1){
                jarak = Vector3.Distance(path.corners[i],
path.corners[i + 1]);
                total += jarak;
                line.SetPosition(i, path.corners[i]);
            }
            total = Mathf.Ceil(total);
            jarakObject.transform.localScale = new Vector3(1,
1, 1);
            jarakValue.text = total.ToString();
            jarakValue.transform.localScale = new Vector3(1,
1, 1);
        }
        else{
            agent.CalculatePath(lift[playerFloor].transform.position, path);
            lastGameObject = GameObject.Instantiate(penanda,
transform.position, Quaternion.identity) as GameObject;
            lineLama = lastGameObject;
            line =
            lastGameObject.GetComponent<LineRenderer>();
            line.SetVertexCount(path.corners.Length);
            jarak =
            Vector3.Distance(lastGameObject.transform.position,
path.corners[0]);
            total = jarak;
            for (int i = 0; i < path.corners.Length; i++){
                if (i + 1 <= path.corners.Length - 1){
                    jarak = Vector3.Distance(path.corners[i],
path.corners[i + 1]);
                    total += jarak;
                    line.SetPosition(i, path.corners[i]);
                }
            }
            lift[destinationFloor].GetComponent<NavMeshAgent>().CalculatePath(
tujuan.transform.position, path);
            lastGameObject = GameObject.Instantiate(penanda,
transform.position, Quaternion.identity) as GameObject;
            lineLama2 = lastGameObject;

```

```

        line2
lastGameObject.GetComponent<LineRenderer>();
        line2.SetVertexCount(path.corners.Length);
        jarak
Vector3.Distance(lastGameObject.transform.position,
path.corners[0]);
        total += jarak;
        for (int i = 0; i < path.corners.Length; i++){
            if (i + 1 <= path.corners.Length - 1){
                jarak = Vector3.Distance(path.corners[i],
path.corners[i + 1]);
                total += jarak;}
            line2.SetPosition(i, path.corners[i]);}
        total = Mathf.Ceil(total);
        jarakObject.transform.localScale = new Vector3(1,
1, 1);
        jarakValue.text = total.ToString();
        jarakValue.transform.localScale = new Vector3(1,
1, 1);}
    }
}
}

```

Kode Sumber 4.14 Implementasi Fitur *Navigation(path)* pada Lantai yang Berbeda Melalui Lift

Untuk perpindahan objek pengguna pada lift, karena objek pengguna didefinisikan sebagai *NavMeshAgent* dan hanya dapat bergerak pada area yang didefinisikan sebagai *Navigation Mesh*, maka untuk perpindahan objek pengguna memanfaatkan fungsi *Warp* yang disediakan fitur *NavMesh*. Kode untuk perpindahan objek pengguna pada lift dapat dilihat pada Kode Sumber 4.15.

```

public class liftScript : MonoBehaviour {
    NavMeshAgent agent;
    GameObject liftlt2_pointer;
    GameObject liftlt1_pointer;
    GameObject liftlt3_pointer;
    void Start()
    {
        agent = gameObject.GetComponent<NavMeshAgent>();
        liftlt1_pointer =
GameObject.FindGameObjectWithTag("liftlt1_pointer");
        liftlt2_pointer =
GameObject.FindGameObjectWithTag("liftlt2_pointer");

```

```

        liftlt3_pointer =
GameObject.FindGameObjectWithTag("liftlt3_pointer");
    }
    public void TransformLantai1()
    {
        agent.Warp(liftlt1_pointer.transform.position);
    }
    public void TransformLantai2()
    {
        agent.Warp(liftlt2_pointer.transform.position);
    }
    public void TransformLantai3()
    {
        agent.Warp(liftlt3_pointer.transform.position);
    }
}

```

Kode Sumber 4.15 Kode Untuk Perpindahan Objek Pengguna pada Lift

4.6.5 Implementasi Fitur *Navigation(Auto)*

Fitur *Navigation(Path)* adalah fitur untuk melakukan navigasi dengan langsung digerakkan oleh sistem menuju destinasi. Apabila saat pemilihan jenis pilihan navigasi pengguna memilih menggunakan fitur *Navigation(Auto)*, maka sistem akan langsung menggerakkan objek pengguna menuju destinasi yang dituju secara *automatic*. Ketika objek pengguna sudah sampai pada destinasi, objek pengguna akan berhenti dan dapat kembali menelusuri lingkungan virtual mall. Potongan kode sumber untuk implementasi fitur *Navigation(Auto)* dapat dilihat pada Kode Sumber 4.16.

```

void AutoPath()
{
    if(liftCase == true)
    {
        if(playerFloor == destinationFloor)
        {
            agent.SetDestination(tujuan.transform.position);
        }
        else
        {
            agent.SetDestination(lift[playerFloor].transform.position);
        }
    }
}

```

```

    }
    else
    {
        agent.SetDestination(tujuan.transform.position);
    }
}

```

Kode Sumber 4.16 Implementasi Fitur Navigation(Auto)

Untuk permasalahan destinasi pada lantai yang berbeda, apabila menggunakan rute eskalator maka diperlukan pengecekan posisi pengguna dan posisi destinasi, kemudian dilanjutkan dengan implementasi fungsi eskalator, sama seperti pada fitur *Navigation(path)*. Sedangkan apabila menggunakan rute lift, objek pengguna akan digerakkan menuju lift terdekat, kemudian berpindah ke lift destinasi, dan dilanjutkan bergerak menuju destinasi. Perpindahan objek pengguna pada lift memanfaatkan *collider* yang ada pada lift. Potongan kode sumber untuk implementasi perpindahan lift pada fitur *Navigation(auto)* dapat dilihat pada Kode Sumber 4.17.

```

void OnTriggerEnter(Collider other)
{
    if(navigatePath == 2)
    {
        if(other.gameObject.tag == tujuan.gameObject.tag)
        {
            ResetPath();
            print(tujuan.transform.position);
        }
        else if(other.gameObject.tag ==
lift[playerFloor].gameObject.tag)
        {
            agent.Warp(lift[destinationFloor].transform.position);
            playerFloor = destinationFloor;
            agent.SetDestination(tujuan.transform.position);
            StartCoroutine("Jeda");
        }
    }
}

```

Kode Sumber 4.17 Implementasi Perpindahan Lift pada Fitur Navigation(Auto)

4.7 Implementasi Skenario Tutorial

Pada tahap ini akan dijelaskan proses dari implementasi skenario tutorial. Skenario tutorial diperlukan untuk memberikan gambaran kepada pengguna tentang fitur-fitur yang ada pada aplikasi. Selain itu skenario tutorial juga dapat berfungsi sebagai pencatatan waktu tiap *task* yang dilakukan pengguna dan akan digunakan untuk pengujian aplikasi. Total *task* yang ada pada tutorial ada 10 *task* seperti yang dijelaskan pada bagian perancangan. Pada pandangan depan pengguna terdapat panel yang menunjukkan kasus yang harus dilakukan pengguna. Tampilan skenario tutorial dapat dilihat pada Gambar 4.22. Potongan kode sumber untuk implementasi skenario tutorial dapat dilihat pada Kode Sumber 4.18.



Gambar 4.22 Tampilan Skenario Tutorial

```
using UnityEngine;
using System.Collections;
using UnityEngine.UI;

public class TutorialScript : MonoBehaviour {
    NavMeshAgent agent;
    public Text tutorialText;
    public GameObject panelTutorial;
    private Autowalk autowalk;
    private Button floorCanvas;
    public CardboardReticle reticle;
    int tutorial = 0;
```

```

//task 1 cari menu navigasi
float waktuMulai;
float waktuBerhenti;
int menit;
int detik;
void Start ()
{
    agent = gameObject.GetComponent<NavMeshAgent>();
    autowalk = GameObject.FindObjectOfType<Autowalk>();
    autowalk.enabled = false;
    floorCanvas= gameObject.GetComponent<Button>();
    task2 = false;
    task3 = false;
    task4 = false;
    task5 = false;
    task6 = false;
    task7 = false;
    task8 = false;
    task9 = false;
    task10 = false;
    tutorialText.text = "Selamat datang di aplikasi realitas
virtual mall";
    StartCoroutine("Intro");
}
IEnumerator Intro()
{
    yield return new WaitForSeconds(3);
    tutorialText.text = "Pertama-tama silahkan cari menu
navigasi. Menu navigasi berada di bawah anda";
    yield return new WaitForSeconds(5);
    tutorialText.text = "3";
    yield return new WaitForSeconds(1);
    tutorialText.text = "2";
    yield return new WaitForSeconds(1);
    tutorialText.text = "1";
    yield return new WaitForSeconds(1);
    tutorialText.text = "Cari menu navigasi";
    tutorial = 1;
}
void Update ()
{
    print(tutorial);
    if (tutorial == 1)
    {
        waktuMulai += Time.deltaTime;
        print("waktumulai" + waktuMulai);
        if (reticle.target != null && reticle.target.tag ==
"floor_canvas")

```

```

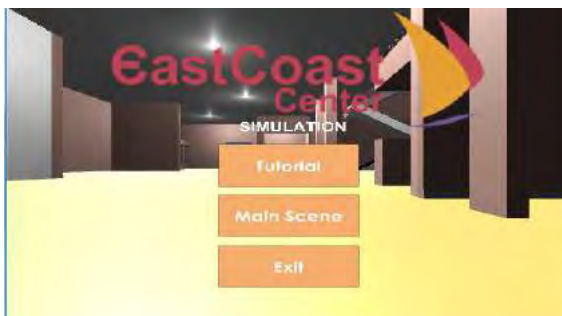
        {
            tutorial = 0;
            waktuBerhenti = waktuMulai;
            print(waktuBerhenti);
            detik = (int) (waktuBerhenti % 60);
            menit = (int) ((waktuBerhenti / 60) % 60);
            tutorialText.text = "Anda menempuh dalam waktu " +
menit.ToString() + " menit " + detik.ToString() + " detik ";
            StartCoroutine("Task2");
        }
    }
}

```

Kode Sumber 4.18 Implementasi Skenario Tutorial

4.8 Implementasi Skenario Menu Utama

Pada skenario menu utama, terdapat logo East Coast Center – Pakuwon yang menandakan bahwa aplikasi ini adalah lingkungan virtual mall tersebut, dan terdapat tiga buah tombol yaitu tombol “Tutorial”, tombol ”Main Scene”, dan tombol “Quit”. Tombol “Tutorial” digunakan untuk berpindah ke skenario tutorial, tombol “Main Scene” digunakan untuk berpindah ke skenario utama, sedangkan tombol “Quit” digunakan untuk keluar dari aplikasi. Tampilan menu utama dapat dilihat pada Gambar 4.23. Potongan kode sumber untuk implementasi perpindahan *scene* dapat dilihat pada Kode Sumber 4.19.



Gambar 4.23 Tampilan Menu Utama

```
using UnityEngine;
using System.Collections;

public class MainMenu : MonoBehaviour {
    public void LoadNewScene(int scene)
    {
        Application.LoadLevel(scene);
    }
    public void Exit()
    {
        Application.Quit();
    }
}
```

Kode Sumber 4.19 Implementasi Perpindahan Scene

BAB V

PENGUJIAN DAN EVALUASI

Pada bab ini dibahas mengenai pengujian dan evaluasi dari implementasi yang telah dilakukan.

5.1 Lingkungan Pengujian

Lingkungan uji coba yang digunakan adalah sebuah smartphone dengan spesifikasi sebagai berikut:

1. Perangkat Keras
 - Tipe : Xiaomi Mi4i
 - Prosesor : Qualcomm Snapdragon 615, Octa-Core 64-bit ARM Cortex A-53
 - Memory : 2048 MB
2. Perangkat Lunak
 - Sistem Operasi : Android OS, v5.0.2 (Lollipop)
3. Perangkat Pendukung
 - Google Cardboard v2

5.2 Pengujian Aplikasi

Pengujian aplikasi dilakukan untuk mengetahui kesesuaian keluaran dari tiap penggunaan fitur terhadap skenario yang dipersiapkan. Berikut ini penjabaran skenario dan hasil uji coba yang dilakukan terhadap perangkat lunak yang dibangun.

5.2.1 Pengujian Fungsionalitas

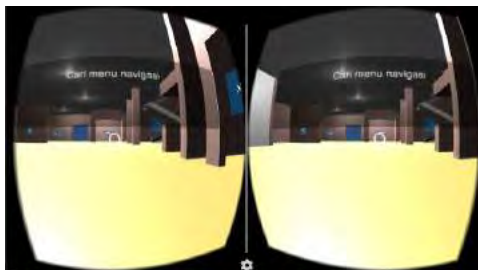
Pada subbab ini dijelaskan beberapa skenario uji coba perangkat lunak secara mandiri berdasarkan metode *blackbox testing* sebagai tolak ukur keberhasilan. Pengujian aplikasi menggunakan skenario tutorial dengan penambahan pencatatan waktu yang ada pada aplikasi. Pengujian yang dilakukan mengacu pada fitur yang disediakan oleh aplikasi. Tiap skenario pengujian mengacu pada tiap kasus penggunaan yang ada pada aplikasi.

5.2.1.1 Pengujian Mencari Menu Navigasi

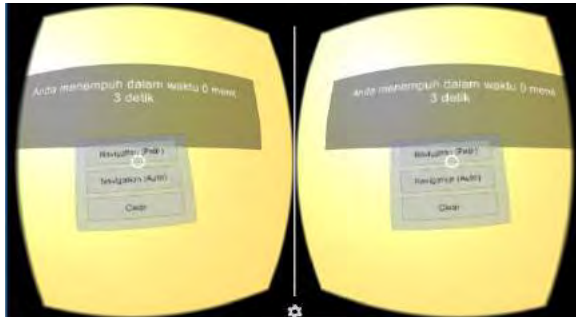
Pengujian mencari menu navigasi merupakan pengujian terhadap pengguna untuk mengetahui apakah pengguna dapat menemukan menu navigasi yang terdapat pada aplikasi. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.1. Tampilan kondisi awal dan kondisi akhir dari skenario pengujian dapat dilihat pada Gambar 5.1 dan Gambar 5.2.

Tabel 5.1 Pengujian Mencari Menu Navigasi

ID	SP-0001
Referensi Kasus Penggunaan	UC-0002
Nama	Mencari Menu Navigasi
Tujuan	Mengetahui apakah pengguna aplikasi dapat menemukan menu navigasi yang terdapat pada aplikasi
Kondisi Awal	Pengguna sudah memilih menu tutorial dan berada di titik awal skenario tutorial
Skenario	Pengguna diminta untuk mencari menu navigasi yang terdapat pada aplikasi
Kondisi Akhir	<ol style="list-style-type: none"> 1. Menu navigasi ditemukan. 2. Aplikasi mencatat waktu pencapaian pengguna.
Hasil Uji Coba	Berhasil
Waktu Pencapaian	1 detik



Gambar 5.1 Kondisi Awal Pengujian Mencari Menu Navigasi



Gambar 5.2 Kondisi Akhir Pengujian Mencari Menu Navigasi

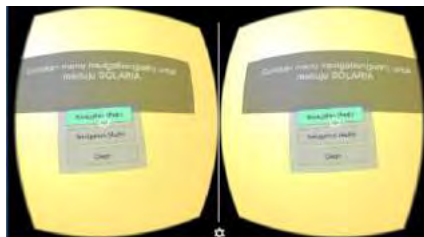
5.2.1.2 Pengujian Menggunakan Fitur *Navigation (Path)* Untuk Mencari Lokasi Dalam Lantai yang Sama

Pengujian menggunakan fitur *Navigation (path)* untuk mencari lokasi dalam lantai yang sama merupakan pengujian terhadap pengguna untuk mengetahui apakah pengguna dapat menggunakan fitur *Navigation(path)* yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang sama dengan posisi pengguna. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.2. Tampilan kondisi awal dan kondisi akhir dari skenario pengujian dapat dilihat pada Gambar 5.3 dan Gambar 5.4.

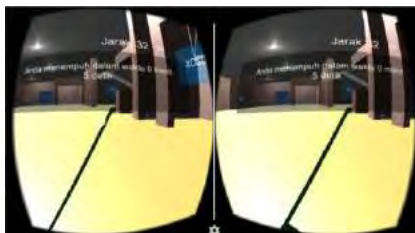
Tabel 5.2 Pengujian Menggunakan Fitur *Navigation (Path)* Untuk Mencari Lokasi Dalam Lantai yang Sama

ID	SP-0002
Referensi Kasus Penggunaan	UC-0003, UC-0004, UC-0005, UC-0006
Nama	Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Sama
Tujuan	Mengetahui apakah pengguna aplikasi dapat menggunakan fitur <i>Navigation(path)</i> yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang sama dengan posisi pengguna

Kondisi Awal	Pengguna berada dalam menu tutorial dan sudah melewati pengujian mencari menu navigasi
Skenario	Pengguna diminta untuk menggunakan fitur <i>Navigation(path)</i> untuk mencari lokasi dalam lantai yang sama dengan posisi pengguna sekarang. Karena destinasi berada pada lantai yang sama, maka pada menu pemilihan rute bebas menggunakan rute eskalator atau lift. Tujuan destinasi sudah ditentukan oleh aplikasi, yaitu menuju lokasi yang bernama “SOLARIA”
Kondisi Akhir	<ol style="list-style-type: none"> 1. <i>Path</i> terbentuk 2. Jarak yang harus ditempuh pengguna menuju lokasi terlihat 3. Aplikasi mencatat waktu pencapaian pengguna
Hasil Uji Coba	Berhasil
Waktu Pencapaian	5 detik



Gambar 5.3 Kondisi Awal Pengujian Menggunakan Fitur *Navigation (Path)* Dalam Lantai yang Sama



Gambar 5.4 Kondisi Akhir Pengujian Menggunakan Fitur *Navigation (Path)* Dalam Lantai yang Sama

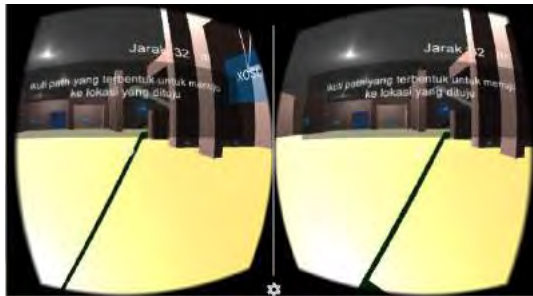
5.2.1.3 Pengujian Mengikuti *Path* yang Terbentuk Untuk Menuju Lokasi Dalam Lantai yang Sama

Pengujian mengikuti *path* yang terbentuk untuk menuju lokasi dalam lantai yang sama merupakan pengujian terhadap pengguna untuk mengetahui apakah pengguna dapat mengikuti *path* yang terbentuk oleh fitur *Navigation(path)* untuk dapat menuju lokasi yang dipilih dalam lantai yang sama dengan posisi pengguna. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.3. Tampilan kondisi awal dan kondisi akhir dari skenario pengujian dapat dilihat pada Gambar 5.5 dan Gambar 5.6.

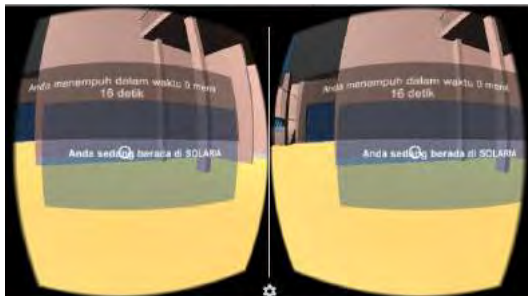
Tabel 5.3 Pengujian Mengikuti *Path* yang Terbentuk Untuk Menuju Lokasi Dalam Lantai yang Sama

ID	SP-0003
Referensi Kasus Penggunaan	UC-0001, UC-0006, UC-0007
Nama	Mengikuti <i>Path</i> yang Terbentuk Untuk Menuju Lokasi Dalam Lantai yang Sama
Tujuan	Mengetahui apakah pengguna dapat mengikuti <i>path</i> yang terbentuk oleh fitur <i>Navigation(path)</i> untuk dapat menuju lokasi yang dipilih dalam lantai yang sama dengan posisi pengguna.
Kondisi Awal	Pengguna berada dalam menu tutorial dan sudah melewati pengujian menggunakan fitur <i>Navigation(path)</i> dalam lantai yang sama. Lokasi yang dituju adalah “SOLARIA”.
Skenario	Pengguna diminta untuk mengikuti <i>path</i> yang terbentuk oleh fitur <i>Navigation(path)</i> untuk dapat menuju lokasi yang dipilih dalam lantai yang sama dengan pengguna.
Kondisi Akhir	<ol style="list-style-type: none"> 1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul 3. Aplikasi mencatat waktu pencapaian

	pengguna
Hasil Uji Coba	Berhasil
Waktu Pencapaian	15 detik



Gambar 5.5 Kondisi Awal Pengujian Mengikuti *Path* yang Terbentuk Untuk Menuju Lokasi Dalam Lantai yang Sama



Gambar 5.6 Kondisi Akhir Pengujian Mengikuti *Path* yang Terbentuk Untuk Menuju Lokasi Dalam Lantai yang Sama

5.2.1.4 Pengujian Menggunakan Fitur Navigation (*Path*) Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator

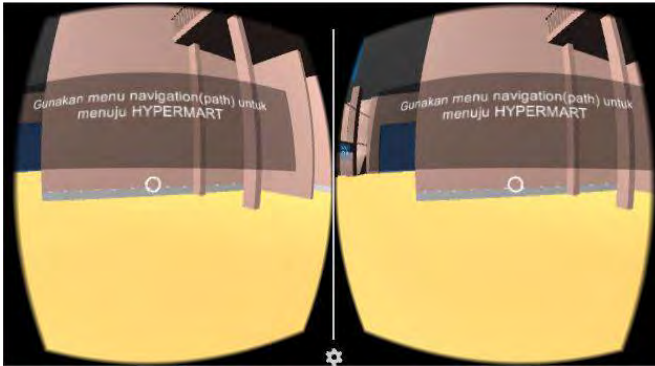
Pengujian menggunakan fitur *Navigation(path)* untuk mencari lokasi dalam lantai yang berbeda melewati eskalator merupakan pengujian terhadap pengguna untuk mengetahui apakah pengguna dapat menggunakan fitur *Navigation(path)* yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang

berbeda dengan posisi pengguna dengan menggunakan rute eskalator. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.4. Tampilan kondisi awal dan kondisi akhir dari skenario pengujian dapat dilihat pada Gambar 5.7 dan Gambar 5.8.

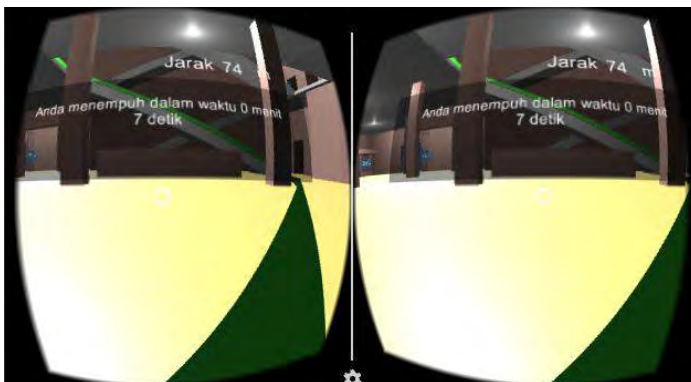
Tabel 5.4 Pengujian Menggunakan Fitur Navigation (Path) Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator

ID	SP-0004
Referensi Kasus Penggunaan	UC-0003, UC-0004, UC-0005, UC-0006
Nama	Menggunakan Fitur Navigation (Path) Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator
Tujuan	Mengetahui apakah pengguna dapat menggunakan fitur <i>Navigation(path)</i> yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang berbeda dengan posisi pengguna dengan menggunakan rute eskalator
Kondisi Awal	Pengguna berada dalam menu tutorial dan sudah melewati pengujian mengikuti <i>path</i> yang terbentuk untuk menuju lokasi dalam lantai yang sama
Skenario	Pengguna diminta untuk menggunakan fitur <i>Navigation(path)</i> untuk mencari lokasi dalam lantai yang berbeda dengan posisi pengguna sekarang dengan menggunakan rute eskalator. Karena destinasi berada pada lantai yang berbeda, maka pada menu pemilihan rute pengguna diminta menggunakan rute eskalator. Tujuan destinasi sudah ditentukan oleh aplikasi, yaitu menuju lokasi yang bernama "HYPERMART"
Kondisi Akhir	<ol style="list-style-type: none"> 1. <i>Path</i> terbentuk 2. Jarak yang harus ditempuh pengguna menuju lokasi terlihat

	3. Aplikasi mencatat waktu pencapaian pengguna
Hasil Uji Coba	Berhasil
Waktu Pencapaian	7 detik



Gambar 5.7 Kondisi Awal Pengujian Menggunakan Fitur Navigation (Path) Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator



Gambar 5.8 Kondisi Akhir Pengujian Menggunakan Fitur Navigation (Path) Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator

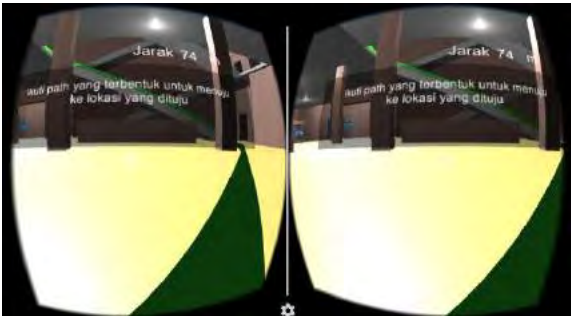
5.2.1.5 Pengujian Mengikuti *Path* Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Eskalator

Pengujian mengikuti *path* yang terbentuk untuk menuju lokasi dalam lantai yang berbeda melewati eskalator merupakan pengujian terhadap pengguna untuk mengetahui apakah pengguna dapat mengikuti *path* yang terbentuk oleh fitur *Navigation(path)* untuk dapat menuju lokasi yang dipilih dalam lantai yang berbeda dengan posisi pengguna dengan menggunakan rute eskalator. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.5. Tampilan kondisi awal dan kondisi akhir dari skenario pengujian dapat dilihat pada Gambar 5.9 dan Gambar 5.10.

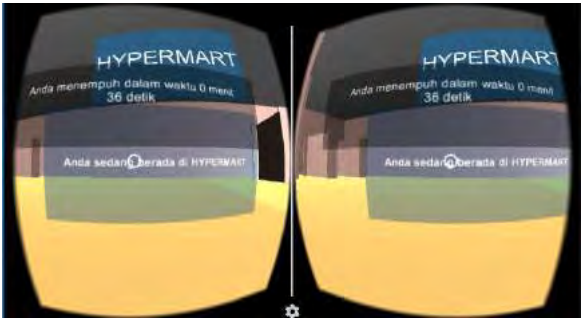
Tabel 5.5 Pengujian Mengikuti *Path* Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Eskalator

ID	SP-0005
Referensi Kasus Penggunaan	UC-0001, UC-0006, UC-0007
Nama	Mengikuti <i>Path</i> Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Eskalator
Tujuan	Mmengetahui apakah pengguna dapat mengikuti <i>path</i> yang terbentuk oleh fitur <i>Navigation(path)</i> untuk dapat menuju lokasi yang dipilih dalam lantai yang berbeda dengan posisi pengguna dengan menggunakan rute eskalator
Kondisi Awal	Pengguna berada dalam menu tutorial dan sudah melewati pengujian menggunakan fitur <i>Navigation(path)</i> untuk mencari lokasi dalam lantai yang berbeda melewati eskalator
Skenario	Pengguna diminta untuk mengikuti <i>path</i> yang terbentuk oleh fitur <i>Navigation(path)</i> untuk dapat menuju lokasi yang dipilih dalam lantai yang berbeda dengan posisi pengguna menggunakan rute eskalator. Lokasi yang dituju

	adalah “HYPERMART”.
Kondisi Akhir	<ol style="list-style-type: none">1. Pengguna sampai pada lokasi yang dituju2. Notifikasi yang menandakan informasi lokasi yang dituju muncul3. Aplikasi mencatat waktu pencapaian pengguna
Hasil Uji Coba	Berhasil
Waktu Pencapaian	35 detik



Gambar 5.9 Kondisi Awal Pengujian Mengikuti *Path* Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Eskalator



Gambar 5.10 Kondisi Akhir Pengujian Mengikuti *Path* Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Eskalator

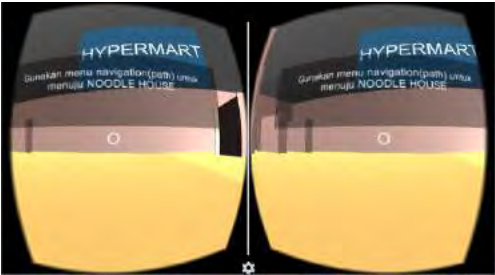
5.2.1.6 Pengujian Menggunakan Fitur *Navigation (Path)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift

Pengujian menggunakan fitur *Navigation (Path)* untuk mencari lokasi dalam lantai yang berbeda melewati lift merupakan pengujian terhadap pengguna untuk mengetahui apakah pengguna dapat menggunakan fitur *Navigation(path)* yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang berbeda dengan posisi pengguna dengan menggunakan rute lift. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.6. Tampilan kondisi awal dan kondisi akhir dari skenario pengujian dapat dilihat pada Gambar 5.11 dan Gambar 5.12.

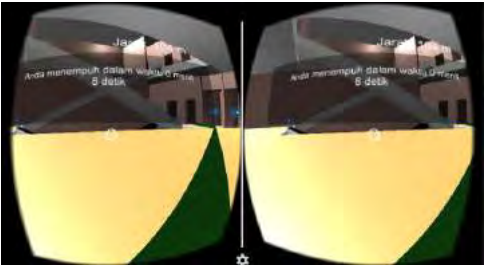
Tabel 5.6 Pengujian Menggunakan Fitur *Navigation (Path)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift

ID	SP-0006
Referensi Kasus Penggunaan	UC-0003, UC-0004, UC-0005, UC-0006
Nama	Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift
Tujuan	Mengetahui apakah pengguna dapat menggunakan fitur <i>Navigation(path)</i> yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang berbeda dengan posisi pengguna dengan menggunakan rute lift
Kondisi Awal	Pengguna berada dalam menu tutorial dan sudah melewati pengujian mengikuti <i>path</i> yang terbentuk untuk menuju lokasi dalam lantai yang berbeda melewati eskalator
Skenario	Pengguna diminta untuk menggunakan fitur <i>Navigation(path)</i> untuk mencari lokasi dalam lantai yang berbeda dengan posisi pengguna sekarang dengan menggunakan rute lift. Karena destinasi berada pada lantai yang berbeda, maka

	pada menu pemilihan rute pengguna diminta menggunakan rute lift. Tujuan destinasi sudah ditentukan oleh aplikasi, yaitu menuju lokasi yang bernama “NOODLE HOUSE”
Kondisi Akhir	<ol style="list-style-type: none">1. <i>Path</i> terbentuk2. Jarak yang harus ditempuh pengguna menuju lokasi terlihat3. Aplikasi mencatat waktu pencapaian pengguna
Hasil Uji Coba	Berhasil
Waktu Pencapaian	8 detik



Gambar 5.11 Kondisi Awal Pengujian Menggunakan Fitur *Navigation (Path)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift



Gambar 5.12 Kondisi Akhir Pengujian Menggunakan Fitur *Navigation (Path)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift

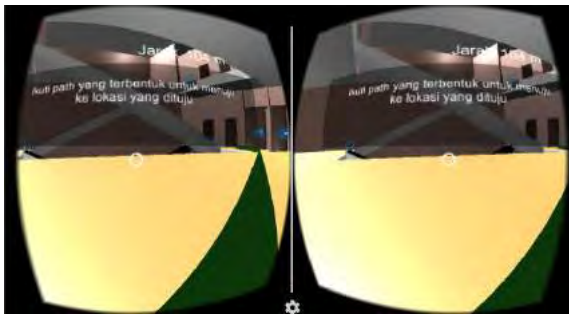
5.2.1.7 Pengujian Mengikuti *Path* Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Lift

Pengujian mengikuti *path* yang terbentuk untuk menuju lokasi dalam lantai yang berbeda melewati lift merupakan pengujian terhadap pengguna untuk mengetahui apakah pengguna dapat mengikuti *path* yang terbentuk oleh fitur *Navigation(path)* untuk dapat menuju lokasi yang dipilih dalam lantai yang berbeda dengan posisi pengguna dengan menggunakan rute lift. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.7. Tampilan kondisi awal dan kondisi akhir dari skenario pengujian dapat dilihat pada Gambar 5.13 dan Gambar 5.14.

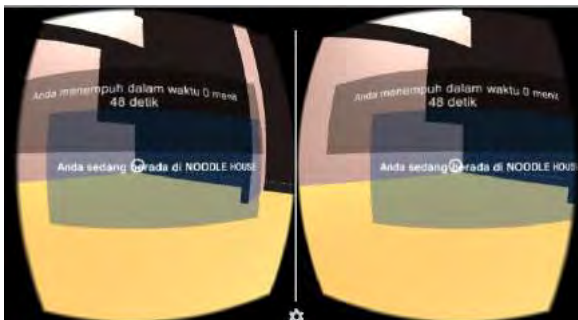
Tabel 5.7 Pengujian Mengikuti *Path* Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Lift

ID	SP-0007
Referensi Kasus Penggunaan	UC-0001, UC-0006, UC-0007
Nama	Mengikuti <i>Path</i> Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Lift
Tujuan	Mengetahui apakah pengguna dapat mengikuti <i>path</i> yang terbentuk oleh fitur <i>Navigation(path)</i> untuk dapat menuju lokasi yang dipilih dalam lantai yang berbeda dengan posisi pengguna dengan menggunakan rute lift
Kondisi Awal	Pengguna berada dalam menu tutorial dan sudah melewati pengujian menggunakan fitur <i>Navigation(path)</i> untuk mencari lokasi dalam lantai yang berbeda melewati lift
Skenario	Pengguna diminta untuk mengikuti <i>path</i> yang terbentuk oleh fitur <i>Navigation(path)</i> untuk dapat menuju lokasi yang dipilih dalam lantai yang berbeda dengan posisi pengguna menggunakan rute lift. Lokasi yang dituju adalah “NOODLE HOUSE”.

Kondisi Akhir	<ol style="list-style-type: none"> 1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul 3. Aplikasi mencatat waktu pencapaian pengguna
Hasil Uji Coba	Berhasil dengan catatan
Waktu Pencapaian	41 detik



Gambar 5.13 Kondisi Awal Mengikuti *Path* Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Lift



Gambar 5.14 Kondisi Akhir Mengikuti *Path* Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Lift

Pada pengujian ini terdapat kendala terkadang tombol untuk menuju ke lantai 1 pada lift susah untuk dipilih. Pengguna harus berada dekat dengan tombol untuk dapat memilih tombol lift.

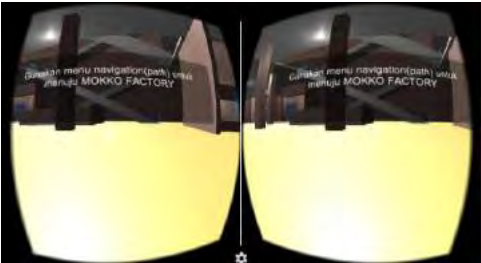
5.2.1.8 Pengujian Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Sama

Pengujian menggunakan fitur *Navigation (Auto)* untuk mencari lokasi dalam lantai yang sama merupakan pengujian terhadap pengguna untuk mengetahui apakah pengguna dapat menggunakan fitur *Navigation(Auto)* yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang sama. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.8. Tampilan kondisi awal dan kondisi akhir dari skenario pengujian dapat dilihat pada Gambar 5.15 dan Gambar 5.16.

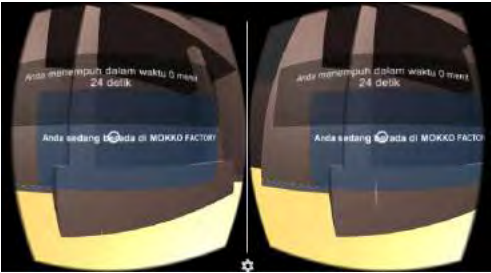
Tabel 5.8 Pengujian Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Sama

ID	SP-0008
Referensi Kasus Penggunaan	UC-0001, UC-0003, UC-0004, UC-0005, UC-0007
Nama	Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Sama
Tujuan	Mengetahui apakah pengguna dapat menggunakan fitur <i>Navigation(Auto)</i> yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang sama
Kondisi Awal	Pengguna berada dalam menu tutorial dan sudah melewati pengujian mengikuti <i>path</i> yang terbentuk untuk menuju lokasi dalam lantai yang berbeda melewati lift
Skenario	Pengguna diminta untuk menggunakan fitur <i>Navigation(auto)</i> untuk mencari lokasi dalam lantai yang sama dengan posisi pengguna sekarang. Karena destinasi berada pada lantai yang sama, maka pada menu pemilihan rute

	bebas menggunakan rute eskalator atau lift. Tujuan destinasi sudah ditentukan oleh aplikasi, yaitu menuju lokasi yang bernama “MOKKO FACTORY”. Setelah destinasi dipilih, sistem akan langsung menggerakkan pengguna menuju destinasi secara <i>automatic</i> .
Kondisi Akhir	<ol style="list-style-type: none">1. Pengguna sampai pada lokasi yang dituju2. Notifikasi yang menandakan informasi lokasi yang dituju muncul3. Aplikasi mencatat waktu pencapaian pengguna
Hasil Uji Coba	Berhasil
Waktu Pencapaian	22 detik



Gambar 5.15 Kondisi Awal Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Sama



Gambar 5.16 Kondisi Akhir Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Sama

5.2.1.9 Pengujian Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator

Pengujian menggunakan fitur *Navigation (Auto)* untuk mencari lokasi dalam lantai yang berbeda melewati eskalator merupakan pengujian terhadap pengguna untuk mengetahui apakah pengguna dapat menggunakan fitur *Navigation(Auto)* yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang berbeda melewati eskalator. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.9. Tampilan kondisi awal dan kondisi akhir dari skenario pengujian dapat dilihat pada Gambar 5.17 dan Gambar 5.18.

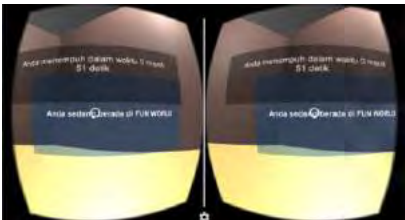
Tabel 5.9 Pengujian Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator

ID	SP-0009
Referensi Kasus Penggunaan	UC-0001, UC-0003, UC-0004, UC-0005, UC-0007
Nama	Menggunakan Fitur Navigation (Auto) Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator
Tujuan	Mengetahui apakah pengguna dapat menggunakan fitur <i>Navigation(Auto)</i> yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang berbeda menggunakan rute eskalator
Kondisi Awal	Pengguna berada dalam menu tutorial dan sudah melewati pengujian menggunakan fitur <i>Navigation(Auto)</i> untuk mencari lokasi dalam lantai yang sama
Skenario	Pengguna diminta untuk menggunakan fitur <i>Navigation(auto)</i> untuk mencari lokasi dalam lantai yang berbeda dengan posisi pengguna sekarang dengan menggunakan rute eskalator. Karena destinasi berada pada lantai yang

	berbeda, maka pada menu pemilihan rute pengguna diminta untuk menggunakan rute eskalator. Tujuan destinasi sudah ditentukan oleh aplikasi, yaitu menuju lokasi yang bernama “FUN WORLD”. Setelah destinasi dipilih, sistem akan langsung menggerakkan pengguna menuju destinasi secara <i>automatic</i> .
Kondisi Akhir	<ol style="list-style-type: none">1. Pengguna sampai pada lokasi yang dituju2. Notifikasi yang menandakan informasi lokasi yang dituju muncul3. Aplikasi mencatat waktu pencapaian pengguna
Hasil Uji Coba	Berhasil
Waktu Pencapaian	49 detik



Gambar 5.17 Kondisi Awal Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator



Gambar 5.18 Kondisi Akhir Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator

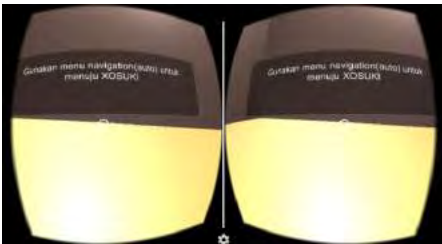
5.2.1.10 Pengujian Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift

Pengujian menggunakan fitur *Navigation (Auto)* untuk mencari lokasi dalam lantai yang berbeda melewati lift merupakan pengujian terhadap pengguna untuk mengetahui apakah pengguna dapat menggunakan fitur *Navigation(Auto)* yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang berbeda melewati lift. Hasil dari pengujian tersebut secara lengkap dapat dilihat pada Tabel 5.10. Tampilan kondisi awal dan kondisi akhir dari skenario pengujian dapat dilihat pada Gambar 5.19 dan Gambar 5.20.

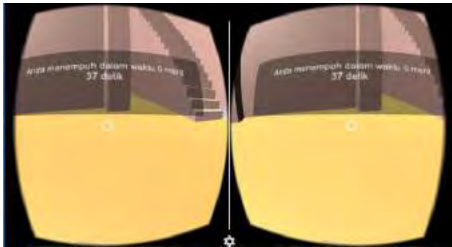
Tabel 5.10 Pengujian Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift

ID	SP-00010
Referensi Kasus Penggunaan	UC-0001, UC-0003, UC-0004, UC-0005, UC-0007
Nama	Menggunakan Fitur Navigation (Auto) Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift
Tujuan	Mengetahui apakah pengguna dapat menggunakan fitur <i>Navigation(Auto)</i> yang terdapat pada aplikasi untuk mencari lokasi dalam lantai yang berbeda menggunakan rute lift
Kondisi Awal	Pengguna berada dalam menu tutorial dan sudah melewati pengujian menggunakan fitur <i>Navigation(Auto)</i> untuk mencari lokasi dalam lantai yang berbeda melewati lift
Skenario	Pengguna diminta untuk menggunakan fitur <i>Navigation(auto)</i> untuk mencari lokasi dalam lantai yang berbeda dengan posisi pengguna sekarang dengan menggunakan rute lift. Karena destinasi berada pada lantai yang berbeda, maka pada menu pemilihan rute pengguna diminta

	untuk menggunakan rute lift. Tujuan destinasi sudah ditentukan oleh aplikasi, yaitu menuju lokasi yang bernama “XOSUKI”. Setelah destinasi dipilih, sistem akan langsung menggerakkan pengguna menuju destinasi secara <i>automatic</i> .
Kondisi Akhir	<ol style="list-style-type: none">1. Pengguna sampai pada lokasi yang dituju2. Notifikasi yang menandakan informasi lokasi yang dituju muncul3. Aplikasi mencatat waktu pencapaian pengguna
Hasil Uji Coba	Berhasil
Waktu Pencapaian	37 detik



Gambar 5.19 Kondisi Awal Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift



Gambar 5.20 Kondisi Akhir Menggunakan Fitur *Navigation (Auto)* Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift

5.2.2 Pengujian Aplikasi Terhadap Pengguna

Aplikasi ini perlu diuji oleh pengguna secara langsung. Hal ini bertujuan untuk mengetahui apakah aplikasi ini sudah membantu pengguna untuk melakukan pencarian lokasi yang ada dalam lingkungan virtual mall. Tujuan lainnya yaitu untuk mengetahui tingkat kemudahan dan kenyamanan aplikasi saat digunakan oleh pengguna. Metode pengujian yang dilakukan pada pengujian aplikasi terhadap pengguna mengacu pada *blackbox testing*.

Pengujian dilakukan oleh 10 orang yang diminta oleh penulis dan bersedia untuk melakukan pengujian. 5 orang yang dipilih adalah orang yang sudah pernah menggunakan perangkat Google Cardboard, serta 5 orang sisanya adalah orang yang belum pernah menggunakan perangkat Google Cardboard. Skenario yang digunakan dalam pengujian adalah skenario tutorial pada aplikasi. Pengguna akan diminta untuk mengoperasikan aplikasi dan menjalankan *task* sesuai rincian *task* yang ada pada lembar *Usability Testing*. Kemudian penulis mencatat waktu pencapaian tiap *task* yang dilakukan pengguna pada lembar *Usability Testing*. Setelah itu pengguna diminta untuk mengisi kuisioner yang telah disediakan. Rincian *task* yang terdapat pada lembar *Usability Testing* dapat dilihat pada Tabel 5.11. Kuisioner yang diberikan berisi pernyataan pengguna saat menggunakan aplikasi dalam aspek kenyamanan fitur, kenyamanan aplikasi, kemudahan aplikasi, dan kegunaan aplikasi. Kuisioner dapat dilihat pada Tabel 5.12. Masing-masing pilihan jawaban pada kuisioner akan memiliki bobot penilaian tersendiri dengan ketentuan sebagai berikut :

- Sangat Tidak Setuju = 1
- Tidak Setuju = 2
- Ragu-ragu = 3
- Setuju = 4
- Sangat Setuju = 5

Tabel 5.11 Rincian *Task* pada *Usability Testing*

Kode	Task	Kondisi Selesai	Waktu Pencapaian
T1	Cari menu navigasi	Menu navigasi ditemukan	
T2	Gunakan fitur “Navigation(path)” untuk mencari lokasi dalam lantai yang sama	1.Path terbentuk 2.Jarak yang harus ditempuh untuk menuju lokasi terlihat	
T3	Ikuti path yang terbentuk sampai menuju ke lokasi yang dituju (lanjutan <i>task</i> nomor 4)	1.Pengguna sampai pada lokasi yang dituju 2.Notifikasi yang menandakan informasi lokasi yang dituju muncul	
T4	Gunakan fitur “Navigation(path)” untuk mencari lokasi dalam lantai yang berbeda melewati eskalator	1.Path terbentuk 2.Jarak yang harus ditempuh untuk menuju lokasi terlihat	
T5	Ikuti path yang terbentuk sampai menuju ke lokasi yang dituju (lanjutan <i>task</i> nomor 6)	1.Pengguna sampai pada lokasi yang dituju 2.Notifikasi yang menandakan informasi lokasi yang dituju muncul	
T6	Gunakan fitur “Navigation(path)” untuk mencari lokasi dalam lantai yang berbeda melewati lift	1.Path terbentuk menuju lift terdekat 2.Jarak yang harus ditempuh untuk menuju	

		lokasi terlihat	
T7	Ikuti path yang terbentuk sampai menuju ke lokasi yang dituju (lanjutan <i>task</i> nomor 8)	1.Pengguna sampai pada lokasi yang dituju 2.Notifikasi yang menandakan informasi lokasi yang dituju muncul	
T8	Gunakan fitur “Navigation(auto)” untuk mencari lokasi dalam lantai yang sama	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	
T9	Gunakan fitur “Navigation(auto)” untuk mencari lokasi dalam lantai yang berbeda melalui eskalator	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	
T10	Gunakan fitur “Navigation(auto)” untuk mencari lokasi dalam lantai yang berbeda melalui lift	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	

Tabel 5.12 Daftar Pernyataan Kuisioner

No	Pernyataan
1	Saya pernah menggunakan Google Cardboard sebelumnya
2	Dalam melakukan pencarian lokasi, saya lebih nyaman menggunakan fitur (<i>Navigation(Path)/Navigation(Auto)/Sama saja</i>)
3	Saya merasa nyaman dalam menggunakan aplikasi ini
4	Saya merasa mudah dalam menjalankan task/ aktivitas yang diberikan

5	Saya merasa seperti berada dalam lingkungan mall saat menggunakan aplikasi ini
6	Saya merasa terbantu dengan adanya sistem navigasi pada aplikasi dalam melakukan pencarian lokasi di dalam lingkungan virtual

5.3 Evaluasi

Tahap evaluasi akan dibagi menjadi dua bagian, yaitu evaluasi pengujian fungsionalitas dan evaluasi pengujian aplikasi terhadap pengguna.

5.3.1 Evaluasi Pengujian Fungsionalitas

Rangkuman mengenai hasil pengujian fungsionalitas dapat dilihat pada Tabel 5.13. Berdasarkan data pada tabel tersebut, dapat disimpulkan bahwa semua skenario pengujian berhasil dijalankan. Sehingga dapat disimpulkan fungsionalitas dari aplikasi dapat bekerja sesuai dengan yang diharapkan.

Tabel 5.13 Rangkuman Hasil Pengujian Fungsionalitas

ID	Nama	Waktu Pencapaian	Hasil
SP-0001	Mencari Menu Navigasi	1 detik	Berhasil
SP-0002	Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Sama	5 detik	Berhasil
SP-0003	Mengikuti <i>Path</i> yang Terbentuk Untuk Menuju Lokasi Dalam Lantai yang Sama	15 detik	Berhasil
SP-0004	Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator	7 detik	Berhasil
SP-0005	Mengikuti <i>Path</i> Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Eskalator	35 detik	Berhasil

SP-0006	Menggunakan Fitur <i>Navigation (Path)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift	8 detik	Berhasil
SP-0007	Mengikuti <i>Path</i> Yang Terbentuk Untuk Menuju Lokasi Dalam Lantai Yang Berbeda Melewati Lift	41 detik	Berhasil
SP-0008	Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Sama	22 detik	Berhasil
SP-0009	Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Eskalator	49 detik	Berhasil
SP-00010	Menggunakan Fitur <i>Navigation (Auto)</i> Untuk Mencari Lokasi Dalam Lantai yang Berbeda Melewati Lift	37 detik	Berhasil

5.3.2 Evaluasi Pengujian Aplikasi Terhadap Pengguna

Pengujian aplikasi dilakukan kepada 10 orang pengguna. 5 orang pengguna memiliki pengalaman menggunakan Google Cardboard, sedangkan 5 orang pengguna lainnya tidak memiliki pengalaman menggunakan Google Cardboard. Rentang umur pengguna adalah 19 sampai 23 tahun dan sedang menempuh pendidikan sebagai mahasiswa. Metode pengambilan kesimpulan dalam evaluasi ini menggunakan metode 5-likert scale.

5.3.2.1 Evaluasi Pengujian *Usability Testing*

Rangkuman hasil *Usability Testing* dari masing-masing pengguna yang memiliki pengalaman menggunakan Google Cardboard dapat dilihat pada Tabel 5.14, sedangkan rangkuman hasil *Usability Testing* dari masing-masing pengguna yang tidak memiliki pengalaman menggunakan Google Cardboard dapat dilihat pada Tabel 5.15. Hasil perbandingan *Usability Testing* dapat dilihat pada Tabel 5.16.

Tabel 5.14 Rangkuman Hasil *Usability Testing A*

Partisipan	Waktu Pencapaian tiap Task (detik)									
	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
Dimas Riskahadi	4	15	19	12	68	16	100	32	62	41
Hafieludin Yusuf Rizana	1	20	17	15	40	18	126	25	69	42
Fajar Setiawan	1	10	17	11	53	19	44	25	50	42
Ikrom Aulia Fahdi	1	11	38	10	36	17	192	35	49	42
Luthfi Soehadak	1	6	16	8	36	7	194	37	54	38
TOTAL	8	62	107	56	233	77	656	154	284	205
RATA-RATA	1.6	12.4	21.4	11.2	46.6	15.4	131.2	30.8	56.8	41

Tabel 5.15 Rangkuman Hasil *Usability Testing B*

Partisipan	Waktu Pencapaian tiap Task (detik)									
	T1	T2	T3	T4	T5	T6	T7	T8	T9	T10
Nusantara Jaya Sakti	1	8	17	11	37	17	153	27	52	36
Zakwan Hilmy	1	8	15	10	41	17	39	29	51	37
Dwiky Okka T	1	31	24	19	37	13	193	34	52	38
Maria Rizqi Yudha S	17	16	70	17	51	15	197	34	61	68
Lydia Angela	3	12	41	35	100	17	117	26	120	54
TOTAL	23	75	167	92	266	79	699	150	336	233
RATA-RATA	4.6	15	33.4	18.4	53.2	15.8	139.8	30	67.2	46.6

Tabel 5.16 Hasil Perbandingan *Usability Testing*

Task	Rata-Rata Waktu Pencapaian (detik)	
	Pernah menggunakan Google Cardboard	Tidak pernah menggunakan Google Cardboard
T1	1.6	4.6
T2	12.4	15
T3	21.4	33.4
T4	11.2	18.4
T5	46.6	53.2
T6	15.4	15.8
T7	131.2	139.8
T8	30.8	30
T9	56.8	67.2
T10	41	46.6

Melihat hasil perbandingan *Usability Testing* dari pengguna yang memiliki pengalaman menggunakan Google Cardboard dan yang belum memiliki pengalaman menggunakan Google Cardboard, dapat disimpulkan bahwa aplikasi lebih mudah digunakan oleh pengguna yang sudah memiliki pengalaman menggunakan Google Cardboard. Hal ini dibuktikan dengan melihat dari waktu pencapaian tiap *task* yang diberikan. Pengguna yang memiliki pengalaman menggunakan Google Cardboard memiliki waktu pencapaian yang lebih cepat dibandingkan dengan pengguna yang tidak memiliki pengalaman menggunakan Google Cardboard pada 9 dari 10 *task* yang diberikan.

5.3.2.2 Evaluasi Hasil Kuisisioner

Jawaban hasil kuisisioner dari masing-masing pengguna dapat dilihat pada Lampiran A. Evaluasi hasil kuisisioner meliputi beberapa aspek yang dirasakan pengguna saat menggunakan aplikasi.

5.3.2.2.1 Evaluasi Pengujian Kenyamanan Fitur

Pengujian kenyamanan fitur dibutuhkan untuk mengetahui fitur manakah yang lebih nyaman digunakan oleh pengguna dari semua fitur yang telah dibuat. Dari hasil pengujian kenyamanan fitur dapat disimpulkan bahwa sebagian besar pengguna lebih nyaman menggunakan fitur *Navigation (Auto)*. Hal ini dibuktikan dengan jawaban yang dipilih oleh pengguna pada kuisioner yang diberikan. Pengguna yang memilih pilihan jawaban *Navigation (Path)* ada 3 orang, pengguna yang memilih pilihan jawaban *Navigation (Auto)* ada 6 orang, dan pengguna yang memilih pilihan jawaban sama saja ada 1 orang.

5.3.2.2.2 Evaluasi Pengujian Kenyamanan Aplikasi

Pengujian kenyamanan aplikasi dibutuhkan untuk mengetahui apakah pengguna sudah merasa nyaman dalam menggunakan aplikasi. Aspek kenyamanan mengacu pada pertanyaan nomor 3 dalam kuisioner. Dari hasil pengujian kenyamanan aplikasi, dapat disimpulkan bahwa sebagian besar pengguna sudah merasa nyaman dalam menggunakan aplikasi. Hal ini dibuktikan dengan jawaban yang dipilih oleh pengguna pada kuisioner yang diberikan. Pengguna yang memilih pilihan jawaban Setuju ada 9 orang, sedangkan pengguna yang memilih pilihan jawaban Tidak Setuju ada 1 orang. Rincian detail hasil kuisioner untuk aspek kenyamanan dapat dilihat pada Tabel 5.17 dan Tabel 5.18.

5.3.2.2.3 Evaluasi Pengujian Kemudahan Aplikasi

Pengujian kemudahan aplikasi dibutuhkan untuk mengetahui apakah pengguna sudah merasa mudah dalam menggunakan aplikasi. Aspek kemudahan mengacu pada pertanyaan nomor 4 dalam kuisioner. Dari hasil pengujian kemudahan aplikasi, dapat disimpulkan bahwa sebagian besar pengguna sudah merasa mudah dalam menggunakan aplikasi. Hal ini dibuktikan dengan jawaban yang dipilih oleh pengguna pada kuisioner yang diberikan. Pengguna yang memilih pilihan jawaban Setuju ada 6

orang, pengguna yang memilih pilihan jawaban Ragu-Ragu ada 3 orang, dan pengguna yang memilih pilihan jawaban Tidak Setuju ada 1 orang. Rincian detail hasil kuisioner untuk aspek kemudahan dapat dilihat pada Tabel 5.17 dan Tabel 5.18.

5.3.2.2.4 Evaluasi Pengujian *Immersivity* Aplikasi

Pengujian kemudahan aplikasi dibutuhkan untuk mengetahui apakah pengguna dapat merasakan seperti berada dalam lingkungan sebenarnya saat menggunakan aplikasi ini. Aspek *immersivity* mengacu pada pertanyaan nomor 5 dalam kuisioner. Dari hasil pengujian kemudahan aplikasi, dapat disimpulkan bahwa sebagian besar pengguna sudah cukup merasakan sensasi seolah-olah berada di lingkungan mall saat menggunakan aplikasi. Hal ini dibuktikan dengan jawaban yang dipilih oleh pengguna pada kuisioner yang diberikan. Pengguna yang memilih pilihan jawaban Setuju ada 4 orang, pengguna yang memilih pilihan jawaban Ragu-Ragu ada 4 orang, pengguna yang memilih pilihan jawaban Tidak Setuju ada 1 orang, dan pengguna yang memilih pilihan jawaban Sangat Setuju ada 1 orang. Rincian detail hasil kuisioner untuk aspek *immersivity* dapat dilihat pada Tabel 5.17 dan Tabel 5.18.

5.3.2.2.5 Evaluasi Pengujian Kegunaan Aplikasi

Pengujian kegunaan aplikasi dibutuhkan untuk mengetahui apakah aplikasi yang telah dibuat membantu pengguna dalam melakukan pencarian lokasi dalam lingkungan virtual. Aspek kegunaan mengacu pada pertanyaan nomor 6 dalam kuisioner. Dari hasil pengujian kegunaan aplikasi, dapat disimpulkan bahwa aplikasi yang telah dibuat berguna dan membantu sebagian besar pengguna dalam melakukan pencarian lokasi di dalam lingkungan virtual. Hal ini dibuktikan dengan jawaban yang dipilih oleh pengguna pada kuisioner yang diberikan. Pengguna yang memilih pilihan jawaban Setuju ada 6 orang, sedangkan pengguna yang memilih pilihan jawaban Sangat Setuju ada 4 orang. Rincian detail hasil kuisioner untuk aspek kegunaan dapat dilihat pada Tabel 5.17 dan Tabel 5.18.

Tabel 5.17 Rangkuman Hasil Kuisioner Aspek Kemudahan, Kenyamanan, Immersivity, dan Kegunaan Aplikasi

Partisipan	Pilihan Jawaban			
	Kenyamanan	Kemudahan	Immersivity	Kegunaan
Dimas Riskahadi	4	4	3	4
Hafieludin Yusuf Rizana	2	2	2	4
Fajar Setiawan	4	4	4	4
Ikrom Aulia Fahdi	4	4	5	5
Luthfi Soehadak	4	4	3	5
Nusantara Jaya Sakti	4	3	4	4
Zakwan Hilmy	4	4	4	4
Dwiky Okka T	4	3	3	5
Maria Rizqi Yudha S	4	3	4	4
Lydia Angela	4	4	3	5

Tabel 5.18 Rangkuman Pilihan Jawaban Aspek Kemudahan, Kenyamanan, Immersivity, dan Kegunaan Aplikasi

Aspek	Skala Pilihan Jawaban				
	1	2	3	4	5
Kenyamanan	0	1	0	9	0
Kemudahan	0	1	3	6	0
Immersivity	0	1	4	4	1
Kegunaan	0	0	0	6	4

5.3.2.3 Evaluasi Kritik dan Saran Pengguna

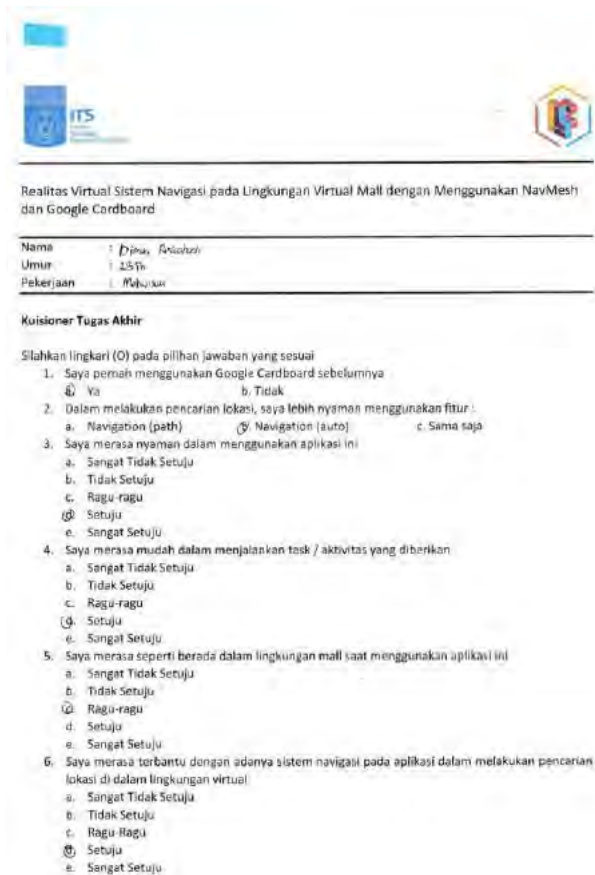
Pada tahap ini akan dipaparkan hasil kritik dan saran yang diberikan oleh pengguna serta evaluasi yang dapat diberikan berdasarkan hasil yang ada. Rangkuman kritik dan saran pengguna dapat dilihat pada Tabel 5.19.

Tabel 5.19 Rangkuman Kritik dan Saran Pengguna

Partisipan	Kritik dan Saran
Dimas Riskahadi	<ul style="list-style-type: none"> • Mungkin assetnya kurang mirip dengan mall sesungguhnya, jadi lebih dilengkapkan • Perpindahan teks terlalu cepat
Hafieludin Yusuf Rizana	Navigasi sudah baik, hanya saja tampilan lebih ditingkatkan atau diperindah untuk menarik pengguna. Program diperingan agar penggunaannya lebih lancar.
Fajar Setiawan	Objek lebih bagus lagi
Ikrom Aulia Fahdi	Asetnya diperbaiki / ditingkatkan supaya lebih terasa real. UI nya dipercantik supaya membuat user nyaman dalam menggunakan aplikasi ini
Luthfi Soehadak	<ul style="list-style-type: none"> • Navigasi menggunakan lift dipermudah • Teks tutorial terlalu cepat ganti
Nusantara Jaya Sakti	Untuk pilihan menu lebih baik mengambang secara 2D daripada di lantai sehingga dapat mudah memilihnya
Zakwan Hilmy	Resolusi ditambah
Dwiky Okka T	Direction tutorial terlalu cepet, butuh waktu tambahan untuk pengguna membaca dan memahami. Overall OK.
Maria Rizqi Yudha S	<ul style="list-style-type: none"> • Tombol buat milih menu susah dipencet • Tombol satu di lift susah dipilihnya
Lydia Angela	Tombolnya dibenerin. Tunggu perintah tutorial jangan kelamaan (kalau bisa diskip)

(Halaman ini sengaja dikosongkan)

LAMPIRAN A KUISIONER



Realitas Virtual Sistem Navigasi pada Lingkungan Virtual Mall dengan Menggunakan NavMesh dan Google Cardboard

Nama : *Dimas Riskahadi*
 Umur : *25th*
 Pekerjaan : *Pelajar*

Kuisisioner Tugas Akhir

Silahkan lingkari (O) pada pilihan jawaban yang sesuai


- Saya pernah menggunakan Google Cardboard sebelumnya
 - ☒ Ya
 - ☐ Tidak
- Dalam melakukan pencarian lokasi, saya lebih nyaman menggunakan fitur :
 - ☐ Navigation (path)
 - ☒ Navigation (auto)
 - ☐ Sama saja
- Saya merasa nyaman dalam menggunakan aplikasi ini
 - ☐ Sangat Tidak Setuju
 - ☐ Tidak Setuju
 - ☐ Ragu-ragu
 - ☒ Setuju
 - ☐ Sangat Setuju
- Saya merasa mudah dalam menjalankan task / aktivitas yang diberikan
 - ☐ Sangat Tidak Setuju
 - ☐ Tidak Setuju
 - ☐ Ragu-ragu
 - ☒ Setuju
 - ☐ Sangat Setuju
- Saya merasa seperti berada dalam lingkungan mall saat menggunakan aplikasi ini
 - ☐ Sangat Tidak Setuju
 - ☐ Tidak Setuju
 - ☒ Ragu-ragu
 - ☐ Setuju
 - ☐ Sangat Setuju
- Saya merasa terbantu dengan adanya sistem navigasi pada aplikasi dalam melakukan pencarian lokasi di dalam lingkungan virtual
 - ☐ Sangat Tidak Setuju
 - ☐ Tidak Setuju
 - ☐ Ragu-Ragu
 - ☒ Setuju
 - ☐ Sangat Setuju

Gambar A.1 Lembar Kuisisioner Dimas Riskahadi

Kritik dan Saran untuk pengembangan selanjutnya

- *Mungkin Asesmen kurang mirip dengan soal sesungguhnya, jadi lebih diungkapkan.
- *penyederhanaan kata terlalu cepat

Surabaya, 10 Juni 2016



(Amun Rukawadi)



Gambar A.2 Lembar Kritik dan Saran Dimas Riskahadi



Usability Testing

No	Task	Kondisi Selesai	Waktu pencapaian
1	Cari menu navigasi	Menu navigasi ditemukan	9
2	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang sama	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	15
3	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 4)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	19
4	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati eskalator	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	12
5	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 6)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	68
6	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati lift	1. Path terbentuk menuju lift terdekat 2. Jarak yang harus ditempuh menuju lokasi terlihat	16
7	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 8)	1. Pengguna sampai pada lokasi yang dituju melewati lift 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	100
8	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang sama	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	32
9	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui eskalator	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui eskalator	62
10	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui lift	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui lift	41

Gambar A.3 Lembar *Usability Testing* Dimas Riskahadi

Realitas Virtual Sistem Navigasi pada Lingkungan Virtual Mall dengan Menggunakan NavMesh dan Google Cardboard


Nama : Hafieludin Yusuf Rizana
 Umur : 20
 Pekerjaan : Mahasiswa

Kuisloner Tugas Akhir

Silahkan lingkari (O) pada pilihan jawaban yang sesuai

- Saya pernah menggunakan Google Cardboard sebelumnya
 - ☒ Ya
 - ☐ Tidak
- Dalam melakukan pencarian lokasi, saya lebih nyaman menggunakan fitur :
 - Navigation (path)
 - ☒ Navigation (auto)
 - Sama saja
- Saya merasa nyaman dalam menggunakan aplikasi ini
 - Sangat Tidak Setuju
 - ☒ Tidak Setuju
 - Ragu-ragu
 - Setuju
 - Sangat Setuju
- Saya merasa mudah dalam menjalankan task / aktivitas yang diberikan
 - Sangat Tidak Setuju
 - ☒ Tidak Setuju
 - Ragu-ragu
 - Setuju
 - Sangat Setuju
- Saya merasa seperti berada dalam lingkungan mall saat menggunakan aplikasi ini
 - Sangat Tidak Setuju
 - ☒ Tidak Setuju
 - Ragu-ragu
 - Setuju
 - Sangat Setuju
- Saya merasa terbantu dengan adanya sistem navigasi pada aplikasi dalam melakukan pencarian lokasi di dalam lingkungan virtual
 - Sangat Tidak Setuju
 - Tidak Setuju
 - Ragu-Ragu
 - ☒ Setuju
 - Sangat Setuju

Gambar A.4 Lembar Kuisloner Hafieludin Yusuf Rizana




Kritik dan Saran untuk pengembangan selanjutnya

Object lebih bagus lagi

Surabaya,2016

(Hafieludin Yusuf Rizana)

Gambar A.5 Lembar Kritik dan Saran Hafieludin Yusuf Rizana



Usability Testing

NO	Task	Kondisi Selesai	Waktu Pencapaian
1	Cari menu navigasi	Menu navigasi ditemukan	1
2	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang sama	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	20
3	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 4)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	17
4	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati eskalator	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	15
5	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 5)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	40
6	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati lift	1. Path terbentuk menuju lift terdekat 2. Jarak yang harus ditempuh menuju lokasi terlihat	18
7	Iikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 8)	1. Pengguna sampai pada lokasi yang dituju melalui lift 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	125
8	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang sama	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	25
9	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui eskalator	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui eskalator	69
10	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui lift	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui lift	42

Gambar A.6 Lembar *Usability Testing* Hafieludin Yusuf Rizana



Realitas Virtual Sistem Navigasi pada Lingkungan Virtual Mali dengan Menggunakan NavMesh dan Google Cardboard



Nama : Fajar Setiawan
Umur : 21 tahun
Pekerjaan : Mahasiswa

Kuisisioner Tugas Akhir

Silahkan lingkari (O) pada pilihan jawaban yang sesuai

1. Saya pernah menggunakan Google Cardboard sebelumnya
 - a. ☒ Ya
 - b. ☐ Tidak
2. Dalam melakukan pencarian lokasi, saya lebih nyaman menggunakan fitur:
 - a. ☐ Navigation (path)
 - b. ☒ Navigation (auto)
 - c. ☐ Sama saja
3. Saya merasa nyaman dalam menggunakan aplikasi ini
 - a. ☐ Sangat Tidak Setuju
 - b. ☐ Tidak Setuju
 - c. ☐ Ragu-ragu
 - d. ☒ Setuju
 - e. ☐ Sangat Setuju
4. Saya merasa mudah dalam menjalankan task / aktivitas yang diberikan
 - a. ☐ Sangat Tidak Setuju
 - b. ☐ Tidak Setuju
 - c. ☐ Ragu-ragu
 - d. ☒ Setuju
 - e. ☐ Sangat Setuju
5. Saya merasa seperti berada dalam lingkungan mali saat menggunakan aplikasi ini
 - a. ☐ Sangat Tidak Setuju
 - b. ☐ Tidak Setuju
 - c. ☐ Ragu-ragu
 - d. ☒ Setuju
 - e. ☐ Sangat Setuju
6. Saya merasa terbantu dengan adanya sistem navigasi pada aplikasi dalam melakukan pencarian lokasi di dalam lingkungan virtual
 - a. ☐ Sangat Tidak Setuju
 - b. ☐ Tidak Setuju
 - c. ☐ Ragu-Ragu
 - d. ☒ Setuju
 - e. ☐ Sangat Setuju


Gambar A.7 Lembar Kuisisioner Fajar Setiawan



Kritik dan Saran untuk pengembangan selanjutnya

.....
navigasi sudah baik hanya saja tampilan lebih ditingkatkan agar diperindah untuk
menarik pengguna. Program diperindah agar penggunaannya lebih lancar
.....
.....
.....
.....

Surabaya, 10 Juni 2016


(FAJAR SETIAWAN)

Gambar A.8 Lembar Kritik dan Saran Fajar Setiawan



Usability Testing

No	Task	Kondisi Sesuai	Waktu penyelesaian
1	Cari menu navigasi	Menu navigasi ditemukan	1
2	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang sama	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	60
3	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 2)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	17
4	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati eskalator	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	11
5	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 6)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	53
6	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati lift	1. Path terbentuk menuju lift terdekat 2. Jarak yang harus ditempuh menuju lokasi terlihat	19
7	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 8)	1. Pengguna sampai pada lokasi yang dituju melewati lift 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	49
8	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang sama	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	25
9	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui eskalator	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui eskalator	50
10	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui lift	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui lift	42

Gambar A.9 Lembar Usability Testing Fajar Setiawan



Realitas Virtual Sistem Navigasi pada Lingkungan Virtual Mall dengan Menggunakan NavMesh dan Google Cardboard

Nama : Ikrom Aulia Fahdi
 Umur : 22
 Pekerjaan : Mahasiswa

Kuisloner Tugas Akhir

Silahkan lingkari (O) pada pilihan jawaban yang sesuai

1. Saya pernah menggunakan Google Cardboard sebelumnya
 - a. ☒ Ya
 - b. ☐ Tidak
2. Dalam melakukan pencarian lokasi, saya lebih nyaman menggunakan fitur :
 - a. Navigation (path)
 - b. ☒ Navigation (auto)
 - c. Sama saja
3. Saya merasa nyaman dalam menggunakan aplikasi ini
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-ragu
 - d. ☒ Setuju
 - e. Sangat Setuju
4. Saya merasa mudah dalam menjalankan task / aktivitas yang diberikan
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-ragu
 - d. ☒ Setuju
 - e. Sangat Setuju
5. Saya merasa seperti berada dalam lingkungan mall saat menggunakan aplikasi ini
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-ragu
 - d. Setuju
 - e. ☒ Sangat Setuju
6. Saya merasa terbantu dengan adanya sistem navigasi pada aplikasi dalam melakukan pencarian lokasi di dalam lingkungan virtual
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-Ragu
 - d. Setuju
 - e. ☒ Sangat Setuju

Gambar A.10 Lembar Kuisloner Ikrom Aulia Fahdi

Kritik dan Saran untuk pengembangan selanjutnya

Asetnya diper baik / ditingkatkan supaya lebih terasa real.

UI juga diperbaiki supaya membuat user nyaman dalam menggunakan aplikasi ini

Surabaya, 10 - 06 - 2016

(From AF)

Gambar A.11 Lembar Kritik dan Saran Ikrom Aulia Fahdi



Usability Testing

No	Task	Kondisi Selesai	Waktu pencapaian
1	Cari menu navigasi	Menu navigasi ditemukan	1
2	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang sama	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	11
3	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 4)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	38
4	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati eskalator	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	10
5	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 6)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	36
6	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati lift	1. Path terbentuk menuju lift terdekat 2. Jarak yang harus ditempuh menuju lokasi terlihat	17
7	Iikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 8)	1. Pengguna sampai pada lokasi yang dituju melalui lift 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	192
8	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang sama	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	35
9	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui eskalator	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui eskalator	49
10	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui lift	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui lift	42

Gambar A.12 Lembar *Usability Testing* Ikrom Aulia Fahdi



Realitas Virtual Sistem Navigasi pada Lingkungan Virtual Mall dengan Menggunakan NavMesh dan Google Cardboard

Nama : Luthfi F. Soehadad
 Umur : 22
 Pekerjaan : Mahasiswa

Kuisloner Tugas Akhir

Silahkan lingkari (O) pada pilihan jawaban yang sesuai

1. Saya pernah menggunakan Google Cardboard sebelumnya
 - ☒ a. Ya
 - ☐ b. Tidak
2. Dalam melakukan pencarian lokasi, saya lebih nyaman menggunakan fitur :
 - a. Navigation (path)
 - ☒ b. Navigation (auto)
 - c. Sama saja
3. Saya merasa nyaman dalam menggunakan aplikasi ini
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-ragu
 - ☒ d. Setuju
 - e. Sangat Setuju
4. Saya merasa mudah dalam menjalankan task / aktivitas yang diberikan
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-ragu
 - ☒ d. Setuju
 - e. Sangat Setuju
5. Saya merasa seperti berada dalam lingkungan mall saat menggunakan aplikasi ini
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - ☒ c. Ragu-ragu
 - d. Setuju
 - e. Sangat Setuju
6. Saya merasa terbantu dengan adanya sistem navigasi pada aplikasi dalam melakukan pencarian lokasi di dalam lingkungan virtual
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-Ragu
 - d. Setuju
 - ☒ e. Sangat Setuju

Gambar A.13 Lembar Kuisloner Luthfi Soehadad

Kritik dan Saran untuk pengembangan selanjutnya

- Navigasi menggunakan kfi dipermudah
- ~~Teks~~ Teks tutorial terlalu cepat ganti

Surabaya, 10 JUN 2016

Luthfi F. Soehadok

Gambar A.14 Lembar Kritik dan Saran Luthfi Soehadok



Usability Testing

No	Task	Kondisi Selesai	Waktu penyelesaian
1	Cari menu navigasi	Menu navigasi ditemukan	1
2	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang sama	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	6
3	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 4)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	16
4	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati eskalator	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	8
5	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 6)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	36
6	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati lift	1. Path terbentuk menuju lift terdekat 2. Jarak yang harus ditempuh menuju lokasi terlihat	7
7	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 8)	1. Pengguna sampai pada lokasi yang dituju melewati lift 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	194
8	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang sama	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	37
9	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui eskalator	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui eskalator	34
10	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui lift	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui lift	28

Gambar A.15 Lembar Usability Testing Luthfi Soehadak




Realitas Virtual Sistem Navigasi pada Lingkungan Virtual Mall dengan Menggunakan NavMesh dan Google Cardboard

Nama	2. Hutanbaca Jaya Sakti
Umur	2020
Pekerjaan	2. Mahasiswa

Kuisisioner Tugas Akhir

Silahkan lingkari (O) pada pilihan jawaban yang sesuai

1. Saya pernah menggunakan Google Cardboard sebelumnya
 - a. Ya ☐
 - b. Tidak ☒
2. Dalam melakukan pencarian lokasi, saya lebih nyaman menggunakan fitur :
 - a. Navigation (path) ☒
 - b. Navigation (auto) ☐
 - c. Sama saja ☐
3. Saya merasa nyaman dalam menggunakan aplikasi ini
 - a. Sangat Tidak Setuju ☐
 - b. Tidak Setuju ☐
 - c. Ragu-ragu ☐
 - d. Setuju ☒
 - e. Sangat Setuju ☐
4. Saya merasa mudah dalam menjalankan task / aktivitas yang diberikan
 - a. Sangat Tidak Setuju ☐
 - b. Tidak Setuju ☐
 - c. Ragu-ragu ☒
 - d. Setuju ☐
 - e. Sangat Setuju ☐
5. Saya merasa seperti berada dalam lingkungan mall saat menggunakan aplikasi ini
 - a. Sangat Tidak Setuju ☐
 - b. Tidak Setuju ☐
 - c. Ragu-ragu ☐
 - d. Setuju ☒
 - e. Sangat Setuju ☐
6. Saya merasa terbantu dengan adanya sistem navigasi pada aplikasi dalam melakukan pencarian lokasi di dalam lingkungan virtual
 - a. Sangat Tidak Setuju ☐
 - b. Tidak Setuju ☐
 - c. Ragu-Ragu ☐
 - d. Setuju ☒
 - e. Sangat Setuju ☐

Gambar A.16 Lembar Kuisisioner Nusantara Jaya Sakti



Kritik dan Saran untuk pengembangan selanjutnya

Untuk bilangan bulat lebih baik menggunakan secara 2D daripada di lantai

Selalu dapat dalam pembelajaran

Surabaya, 4 Juli 2016

Ut

(Nusantara Jaya Sakti)



Gambar A.17 Lembar Kritik dan Saran Nusantara Jaya Sakti



Usability Testing

No	Task	Kondisi Solusi	Waktu pencapaian
1	Cari menu navigasi	Menu navigasi ditemukan	1
2	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang sama	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	8
3	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 4)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	17
4	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati eskalator	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	41
5	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 6)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	37
6	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati lift	1. Path terbentuk menuju lift terdekat 2. Jarak yang harus ditempuh menuju lokasi terlihat	17
7	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 8)	1. Pengguna sampai pada lokasi yang dituju melalui lift 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	153
8	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang sama	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	27
9	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui eskalator	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui eskalator	52
10	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui lift	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui lift	36

Gambar A.18 Lembar *Usability Testing* Nusantara Jaya Sakti

Realitas Virtual Sistem Navigasi pada Lingkungan Virtual Mall dengan Menggunakan NavMesh dan Google Cardboard

Nama : Zakwan Hilmy
Umur : 21
Pekerjaan : Mahasiswa

Kuisisioner Tugas Akhir

Silahkan lingkari (O) pada pilihan jawaban yang sesuai

- Saya pernah menggunakan Google Cardboard sebelumnya
 - ☐ Ya
 - ☒ Tidak
- Dalam melakukan pencarian lokasi, saya lebih nyaman menggunakan fitur :
 - ☐ Navigation (path)
 - ☒ Navigation (auto)
 - ☐ Sama saja
- Saya merasa nyaman dalam menggunakan aplikasi ini
 - ☐ Sangat Tidak Setuju
 - ☐ Tidak Setuju
 - ☐ Ragu-ragu
 - ☒ Setuju
 - ☐ Sangat Setuju
- Saya merasa mudah dalam menjalankan task / aktivitas yang diberikan
 - ☐ Sangat Tidak Setuju
 - ☐ Tidak Setuju
 - ☐ Ragu-ragu
 - ☒ Setuju
 - ☐ Sangat Setuju
- Saya merasa seperti berada dalam lingkungan mall saat menggunakan aplikasi ini
 - ☐ Sangat Tidak Setuju
 - ☐ Tidak Setuju
 - ☐ Ragu-ragu
 - ☒ Setuju
 - ☐ Sangat Setuju
- Saya merasa terbantu dengan adanya sistem navigasi pada aplikasi dalam melakukan pencarian lokasi di dalam lingkungan virtual
 - ☐ Sangat Tidak Setuju
 - ☐ Tidak Setuju
 - ☐ Ragu-Ragu
 - ☒ Setuju
 - ☐ Sangat Setuju

Gambar A.19 Lembar Kuisisioner Zakwan Hilmy

Kritik dan Saran untuk pengembangan selanjutnya

Resolusi ditangguh

Surabaya, 21-6-2016

(Zaidan Umar)

Gambar A.20 Lembar Kritik dan Saran Zakwan Hilmy



Usability Testing

No	Taak	Kondisi Selesai	Waktu pencapaian
1	Cari menu navigasi	Menu navigasi ditemukan	1
2	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang sama	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	8
3	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 4)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	15
4	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati eskalator	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	10
5	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 6)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	41
6	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati lift	1. Path terbentuk menuju lift terdekat 2. Jarak yang harus ditempuh menuju lokasi terlihat	17
7	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 8)	1. Pengguna sampai pada lokasi yang dituju melewati lift 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	39
8	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang sama	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	29
9	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui eskalator	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui eskalator	51
10	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui lift	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui lift	37

Gambar A.21 Lembar *Usability Testing* Zakwan Hilmy




Realitas Virtual Sistem Navigasi pada Lingkungan Virtual Mall dengan Menggunakan NavMesh dan Google Cardboard



Nama : DWIKY OKKA T
 Umur : 20 THN
 Pekerjaan : Mahasiswa

Kuisisioner Tugas Akhir


Silahkan lingkari (O) pada pilihan jawaban yang sesuai

- Saya pernah menggunakan Google Cardboard sebelumnya
 - Ya
 - ☒ Tidak
- Dalam melakukan pencarian lokasi, saya lebih nyaman menggunakan fitur :
 - Navigation (path)
 - Navigation (auto)
 - ☒ Sama saja
- Saya merasa nyaman dalam menggunakan aplikasi ini
 - Sangat Tidak Setuju
 - Tidak Setuju
 - Ragu-ragu
 - ☒ Setuju
 - Sangat Setuju
- Saya merasa mudah dalam menjalankan task / aktivitas yang diberikan
 - Sangat Tidak Setuju
 - Tidak Setuju
 - ☒ Ragu-ragu
 - Setuju
 - Sangat Setuju
- Saya merasa seperti berada dalam lingkungan mall saat menggunakan aplikasi ini
 - Sangat Tidak Setuju
 - Tidak Setuju
 - ☒ Ragu-ragu
 - Setuju
 - Sangat Setuju
- Saya merasa terbantu dengan adanya sistem navigasi pada aplikasi dalam melakukan pencarian lokasi di dalam lingkungan virtual
 - Sangat Tidak Setuju
 - Tidak Setuju
 - Ragu-Ragu
 - Setuju
 - ☒ Sangat Setuju

Gambar A.22 Lembar Kuisisioner Dwiky Okka T

	 ITS Institut Teknologi Sepuluh Nopember	
	<p>Kritik dan Saran untuk pengembangan selanjutnya</p> <p>Direction tutorial terlalu cepat butuh waktu tambahan untuk penguraian membaca & Memahami. Overall OK.</p>	

Surabaya, 4 Juni 2016


(Dwiky OKKA T)

Gambar A.23 Lembar Kritik dan Saran Dwiky Okka T



Usability Testing

No	Task	Kondisi Selesai	Waktu penyelesaian
1	Cari menu navigasi	Menu navigasi ditemukan	1
2	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang sama	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	31
3	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 4)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	24
4	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati eskalator	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	19
5	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 6)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	37
6	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati lift	1. Path terbentuk menuju lift terdekat 2. Jarak yang harus ditempuh menuju lokasi terlihat	13
7	Iikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 8)	1. Pengguna sampai pada lokasi yang dituju melalui lift 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	193
8	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang sama	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	39
9	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui eskalator	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui eskalator	52
10	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui lift	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui lift	38

Gambar A.24 Lembar *Usability Testing* Dwiky Okka T




Realitas Virtual Sistem Navigasi pada Lingkungan Virtual Mall dengan Menggunakan NavMesh dan Google Cardboard

Nama : Maria Rizqi Yudha S
 Umur : 20
 Pekerjaan : Mahasiswa

Kuisloner Tugas Akhir

Silahkan lingkari (O) pada pilihan jawaban yang sesuai

1. Saya pernah menggunakan Google Cardboard sebelumnya
 - a. Ya
 - ☒ b. Tidak
2. Dalam melakukan pencarian lokasi, saya lebih nyaman menggunakan fitur :
 - ☒ a. Navigation (path)
 - b. Navigation (auto)
 - c. Sama saja
3. Saya merasa nyaman dalam menggunakan aplikasi ini
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-ragu
 - ☒ d. Setuju
 - e. Sangat Setuju
4. Saya merasa mudah dalam menjalankan task / aktivitas yang diberikan
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - ☒ c. Ragu-ragu
 - d. Setuju
 - e. Sangat Setuju
5. Saya merasa seperti berada dalam lingkungan mall saat menggunakan aplikasi ini
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-ragu
 - ☒ d. Setuju
 - e. Sangat Setuju
6. Saya merasa terbantu dengan adanya sistem navigasi pada aplikasi dalam melakukan pencarian lokasi di dalam lingkungan virtual
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-Ragu
 - ☒ d. Setuju
 - e. Sangat Setuju

Gambar A.25 Lembar Kuisloner Maria Rizqi Yudha S

Kritik dan Saran untuk pengembangan selanjutnya

Tambahkan lebih banyak gambar

Tambahkan lebih banyak gambar

Surabaya, 4 Juni 2016

(Maria Rizqi Yudha S.)


Gambar A.26 Lembar Kritik dan Saran Maria Rizqi Yudha S



Usability Testing

No	Task	Kondisi Selesai	Waktu pencapaian
1	Cari menu navigasi	Menu navigasi ditemukan	17
2	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang sama	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	16
3	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 4)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	70
4	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati eskalator	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	17
5	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 6)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	51
6	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati lift	1. Path terbentuk menuju lift terdekat 2. Jarak yang harus ditempuh menuju lokasi terlihat	15
7	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 8)	1. Pengguna sampai pada lokasi yang dituju melewati lift 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	197
8	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang sama	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	39
9	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui eskalator	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui eskalator	61
10	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui lift	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui lift	68

Gambar A.27 Lembar Usability Testing Maria Rizqi Yudha S

Realitas Virtual Sistem Navigasi pada Lingkungan Virtual Mall dengan Menggunakan NavMesh dan Google Cardboard

Nama : Lydia Angela
 Umur : 19
 Pekerjaan : Belajar

Kuisisioner Tugas Akhir

Silahkan lingkari (O) pada pilihan jawaban yang sesuai

1. Saya pernah menggunakan Google Cardboard sebelumnya
 - a. Ya
 - b. Tidak
2. Dalam melakukan pencarian lokasi, saya lebih nyaman menggunakan fitur:
 - a. Navigation (path)
 - b. Navigation (auto)
 - c. Sama saja
3. Saya merasa nyaman dalam menggunakan aplikasi ini
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-ragu
 - d. Setuju
 - e. Sangat Setuju
4. Saya merasa mudah dalam menjalankan task / aktivitas yang diberikan
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-ragu
 - d. Setuju
 - e. Sangat Setuju
5. Saya merasa seperti berada dalam lingkungan mall saat menggunakan aplikasi ini
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-ragu
 - d. Setuju
 - e. Sangat Setuju
6. Saya merasa terbantu dengan adanya sistem navigasi pada aplikasi dalam melakukan pencarian lokasi di dalam lingkungan virtual
 - a. Sangat Tidak Setuju
 - b. Tidak Setuju
 - c. Ragu-Ragu
 - d. Setuju
 - e. Sangat Setuju

Gambar A.28 Lembar Kuisisioner Lydia Angela



Kritik dan Saran untuk pengembangan selanjutnya

Tombolnya diberikan

Tinggi permukaan tutulial jao 1018 m

(kalo lg ada strip)

Surabaya, 2016

(*Lydia AG*)

Gambar A.29 Lembar Kritik dan Saran Lydia Angela



Usability Testing

No.	Task	Kondisi Selesai	Waktu pencapaian
1	Cari menu navigasi	Menu navigasi ditemukan	3
2	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang sama	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	12
3	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 4)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	91
4	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati eskalator	1. Path terbentuk 2. Jarak yang harus ditempuh menuju lokasi terlihat	35
5	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 6)	1. Pengguna sampai pada lokasi yang dituju 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	100
6	Gunakan fitur "Navigation (path)" untuk mencari lokasi dalam lantai yang berbeda melewati lift	1. Path terbentuk menuju lift terdekat 2. Jarak yang harus ditempuh menuju lokasi terlihat	17
7	Ikuti path yang terbentuk sampai menuju lokasi yang dituju (lanjutan task nomor 8)	1. Pengguna sampai pada lokasi yang dituju melewati lift 2. Notifikasi yang menandakan informasi lokasi yang dituju muncul	117
8	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang sama	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi	36
9	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui eskalator	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui eskalator	120
10	Gunakan fitur "Navigation (auto)" untuk mencari lokasi dalam lantai yang berbeda melalui lift	Pengguna sampai pada lokasi yang dipilih dengan digerakkan secara otomatis oleh aplikasi melalui lift	54

Gambar A.30 Lembar *Usability Testing* Lydia Angela

BAB VI

KESIMPULAN DAN SARAN

Pada bab ini dijelaskan mengenai kesimpulan yang dapat diambil dari hasil pengujian yang telah dilakukan sebagai jawaban dari rumusan masalah yang dikemukakan. Selain kesimpulan, juga diberi saran untuk pengembangan aplikasi kedepannya.

6.1. Kesimpulan

Dalam proses pengerjaan tugas akhir mulai dari tahap perancangan, implementasi, dan pengujian yang telah dilakukan terhadap aplikasi, didapatkan kesimpulan sebagai berikut:

1. Dengan menggunakan Cardboard SDK for Unity, aplikasi yang telah dibuat dapat terintegrasikan dengan perangkat Google Cardboard. Dengan menyalin data yang berada pada *package* Cardboard., Unity sudah dapat digunakan untuk membuat aplikasi realitas virtual dengan memanfaatkan Google Cardboard sebagai perangkat pendukung.
2. Ditinjau dari hasil *Usability Testing* yang dilakukan oleh penguji, dapat disimpulkan bahwa aplikasi ini lebih mudah digunakan oleh pengguna yang sudah memiliki pengalaman menggunakan Google Cardboard dibandingkan dengan pengguna yang belum memiliki pengalaman menggunakan Google Cardboard.
3. Ditinjau dari hasil evaluasi pengujian aplikasi terhadap pengguna yang telah dilakukan, dapat disimpulkan bahwa sebagian besar pengguna sudah merasa nyaman dan mudah dalam menggunakan aplikasi. Sebagian besar pengguna juga sudah cukup merasakan sensasi seolah-olah berada di lingkungan mall saat menggunakan aplikasi. Aplikasi yang telah dibuat berguna dan membantu sebagian besar pengguna dalam melakukan pencarian lokasi di dalam lingkungan virtual, dan sebagian besar pengguna lebih nyaman

menggunakan fitur *Navigation(Auto)* saat melakukan pencarian lokasi.

4. NavMesh tidak menyediakan fitur untuk membuat navigasi dalam bentuk garis. Perlu dikombinasikan dengan Line Renderer untuk dapat membuat navigasi dalam bentuk garis.
5. NavMesh dapat digunakan untuk membuat navigasi dalam kasus bangunan bertingkat, namun diperlukan penambahan beberapa fungsi agar navigasi yang dihasilkan berjalan dengan benar, seperti penambahan fungsi eskalator dan lift.
6. Dari hasil implementasi dan pengujian yang dilakukan, didapatkan kesimpulan bahwa fitur NavMesh pada Unity3D berhasil digunakan untuk membuat sistem navigasi pada lingkungan virtual mall.

6.2. Saran

Adapun saran yang diberikan untuk mengembangkan aplikasi kedepannya antara lain:

1. Pembuatan sistem navigasi pada lingkungan virtual dapat diterapkan untuk studi kasus lain yang berkaitan dengan bangunan atau lokasi lain seperti rumah sakit, perencanaan kota, taman bermain, dan sebagainya.
2. Untuk mendapatkan hasil visualisasi yang lebih baik, disarankan untuk *smartphone* yang memiliki spesifikasi tinggi, terutama pada bagian GPU (*Graphic Processing Unit*).

DAFTAR PUSTAKA

- [1] Fernandez, Rodrigo Perez. 2014. *Virtual Reality in a Shipbuilding Environment*. Elsevier, Amsterdam
- [2] Blender, "about-blender," Blender, [Online]. Available: <http://www.blender.org/about/>. [Accessed 12 April 2016]
- [3] Nalwan, A. 1998. *Pemrograman Animasi dan Game Profesional*. Elex Media Komputindo, Jakarta
- [4] Unity, "Game engine, tools and multi platfrom," Unity, [Online]. Available: <http://unity3d.com/unity>. [Accessed 12 April 2016].
- [5] Microsoft, "Microsoft studio- development tools," Microsoft, [Online]. Available: <http://www.visualstudio.com/>. [Accessed 12 April 2016].
- [6] "NavMesh," Unity, [Online]. Available: <http://docs.unity3d.com/ScriptReference/NavMesh.html>. [Accessed 15 December 2015].
- [7] X. Cui and H. Shi, "An Overview of Pathfinding in Navigation Mesh," *IJCSNS International Journal of Computer Science and Network Security*, vol. 12, no. 12, 2012.

(Halaman ini sengaja dikosongkan)

BIODATA PENULIS



Wahyu Widyananda, lahir pada tanggal 1 November 1994 di Malang, Jawa Timur. Hobi yang dimiliki adalah menonton film, bermain game, dan belajar hal baru. Penulis menempuh pendidikan mulai dari MIN Malang 1 (2001-2006), MTsN Malang 1 (2006-2009), SMAN 1 Malang (2009-2012), dan Teknik Informatika ITS (2012-2016). Di jurusan Teknik Informatika ITS, penulis mengambil bidang minat Interaksi Grafika dan Seni (IGS) dan memiliki ketertarikan dalam eksplorasi teknologi di bidang *game*, *augmented reality*, *virtual reality*, dan perangkat bergerak. Selama perkuliahan, penulis aktif dalam organisasi kemahasiswaan, antara lain Staf Media Informasi Himpunan Mahasiswa Teknik Computer-Informatika, Staf Syi'ar Keluarga Muslim Informatika, Staf Divisi Perlengkapan dan Transportasi Schematics 2013, serta menjadi Staf Divisi Keamanan dan Transportasi Schematics 2014. Penulis juga aktif dalam UKM ITS Foreign Language Society. Penulis dapat dihubungi melalui surel : wahyuwidyananda123@gmail.com.